

WEBIST 2014

10th International Conference on Web Information
Systems and Technologies

PROCEEDINGS

Volume 2

Barcelona, Spain

3 - 5 April, 2014

Sponsored by:



Technically sponsored by:



WEBIST 2014

Proceedings of the
10th International Conference on
Web Information Systems and Technologies

Volume 2

Barcelona, Spain

3 - 5 April, 2014

Sponsored by
**INSTICC – Institute for Systems and Technologies of Information, Control
and Communication**

Technically Sponsored by
ERCIS – European Research Center for Information Systems

Copyright © 2014 SCITEPRESS – Science and Technology Publications
All rights reserved

Edited by Valérie Monfort and Karl-Heinz Krempels

Printed in Portugal

ISBN: 978-989-758-024-6

Depósito Legal: 372333/14

<http://www.webist.org>
webist.secretariat@insticc.org

BRIEF CONTENTS

| | |
|--|------|
| INVITED SPEAKERS | IV |
| SPECIAL SESSION CHAIR | IV |
| ORGANIZING AND STEERING COMMITTEES | V |
| PROGRAM COMMITTEE | VI |
| AUXILIARY REVIEWERS | X |
| SPECIAL SESSION PROGRAM COMMITTEE | X |
| SELECTED PAPERS BOOK | X |
| FOREWORD | XIII |
| CONTENTS | XV |

INVITED SPEAKERS

Steven Willmott

3scale

Spain

Fabien Gandon

INRIA

France

Andreas Pfeiffer

Huject GmbH - joint venture of BMW Group, Bosch, Daimler, EnBW, RWE & Siemens

Germany

Zakaria Maamar

Zayed University

U.A.E.

SPECIAL SESSION CHAIR

SPECIAL SESSION ON BUSINESS APPS

Tim A. Majchrzak, University of Münster, Germany

ORGANIZING AND STEERING COMMITTEES

CONFERENCE CHAIR

Valérie Monfort, Université de Paris1 Panthéon Sorbonne, France

PROGRAM CHAIR

Karl-Heinz Krempels, RWTH Aachen University, Germany

PROCEEDINGS PRODUCTION

Marina Carvalho, INSTICC, Portugal

Helder Coelhas, INSTICC, Portugal

Bruno Encarnação, INSTICC, Portugal

Ana Guerreiro, INSTICC, Portugal

Andreia Moita, INSTICC, Portugal

Raquel Pedrosa, INSTICC, Portugal

Vitor Pedrosa, INSTICC, Portugal

Cláudia Pinto, INSTICC, Portugal

Sara Santiago, INSTICC, Portugal

Fábio Santos, INSTICC, Portugal

José Varela, INSTICC, Portugal

CD-ROM PRODUCTION

Pedro Varela, INSTICC, Portugal

GRAPHICS PRODUCTION AND WEBDESIGNER

André Lista, INSTICC, Portugal

Mara Silva, INSTICC, Portugal

SECRETARIAT

Carla Mota, INSTICC, Portugal

WEBMASTER

Susana Ribeiro, INSTICC, Portugal

PROGRAM COMMITTEE

Jose Luis Herrero Agustin, University of Extremadura, Spain

Mugurel Ionut Andreica, Polytechnic University of Bucharest, Romania

Guglielmo de Angelis, CNR - IASI, Italy

Margherita Antona, Foundation for Research and Technology - Hellas (FORTH), Greece

Valeria De Antonellis, University of Brescia, Italy

Liliana Ardissono, Universita' Di Torino, Italy

Giuliano Armano, University of Cagliari, Italy

Ismailcem Budak Arpinar, University of Georgia, U.S.A.

Elarbi Badidi, United Arab Emirates University, U.A.E.

Andrea Ballatore, University College Dublin, Ireland

David Bell, Brunel University, U.K.

Orlando Belo, University of Minho, Portugal

Werner Beuschel, CRADL Lab, Germany

Christoph Bussler, VoxeoLabs, Inc., U.S.A.

Maria Claudia Buzzi, CNR, Italy

Elena Calude, Massey University, Institute of Natural and Mathematical Sciences, New Zealand

Pasquina Campanella, University of Bari "Aldo Moro", Italy

Cinzia Cappiello, Politecnico di Milano, Italy

Sven Casteleyn, Universitat Jaume I, Spain

Federica Cena, University of Torino, Italy

Wei Qin Chen, University of Bergen, Norway

Dickson Chiu, Dickson Computer Systems, Hong Kong

Chin-Wan Chung, Korea Advanced Institute of Science and Technology (KAIST), Korea, Republic of

Christophe Claramunt, Naval Academy Research Institute, France

Mihaela Cocca, University of Portsmouth, U.K.

Martine De Cock, Ghent University, Belgium

Christine Collet, Grenoble Institute of Technology, France

Marco Comuzzi, City University London, U.K.

Isabelle Comyn-Wattiau, Cnam & Essec, France

Anna Corazza, University of Naples "Federico II", Italy

Daniel Cunliffe, University of South Wales, U.K.

Florian Daniel, University of Trento, Italy

Mats Daniels, Uppsala University, Sweden

Mark M. Davydov, Independent Consultant, U.S.A.

Steven Demurjian, University of Connecticut, U.S.A.

Enrico Denti, Alma Mater Studiorum - Università di Bologna, Italy

Stefan Dessloch, Kaiserslautern University of Technology, Germany

Oscar Díaz, University of Basque Country, Spain

Josep Domingo-Ferrer, Universitat Rovira i Virgili, Spain

Atilla Elci, Aksaray University, Turkey

Vadim Ermolayev, Zaporozhye National University, Ukraine

Larbi Esmahi, Athabasca University, Canada

Davide Eynard, University of Lugano, Switzerland

Alexander Felfernig, Technische Universität Graz, Austria

Anna Fensel, STI Innsbruck, University of Innsbruck, Austria

Miriam Fernandez, The Open University, U.K.

Joao Carlos Amaro Ferreira, ISEL, Portugal

Josep-Lluís Ferrer-Gomila, Balearic Islands University, Spain

Karla Donato Fook, IFMA - Instituto Federal de Educação Ciência e Tecnologia do Maranhão, Brazil

Geoffrey Charles Fox, Indiana University, U.S.A.

PROGRAM COMMITTEE (CONT.)

Pasi Fränti, Speech and Image Processing Unit,
University of Eastern Finland, Finland

Britta Fuchs, FH Aachen, Germany

Giovanni Fulantelli, Italian National Research
Council, Italy

Ombretta Gaggi, Università di Padova, Italy

John Garofalakis, University of Patras, Greece

Panagiotis Germanakos, University of Cyprus,
Cyprus

Massimiliano Giacomini, Università degli Studi di
Brescia, Italy

José Antonio Gil, Universitat Politècnica de
València, Spain

Anna Goy, University of Torino, Italy

Thomas Greene, M.I.T., U.S.A.

Ratvinder Grewal, Laurentian University, Canada

Begoña Gros, University of Barcelona, Spain

William Grosky, University of Michigan -
Dearborn, U.S.A.

Angela Guercio, Kent State University, U.S.A.

Francesco Guerra, University of Modena and
Reggio Emilia, Italy

Miguel Guinaláiu, Universidad de Zaragoza, Spain

Shanmugasundaram Hariharan, TRP
Engineering College, India

Ioannis Hatzilygeroudis, University of Patras,
Greece

Stylianos Hatzipanagos, King's College London,
U.K.

A. Henten, Aalborg University, Denmark

Emilio Insfran, Universitat Politècnica de
València, Spain

Ivan Ivanov, SUNY Empire State College, U.S.A.

Kai Jakobs, RWTH Aachen University, Germany

Dietmar Jannach, Technical University of
Dortmund, Germany

Monique Janneck, Luebeck University of Applied
Sciences, Germany

Ivan Jelinek, Czech Technical University in
Prague, Czech Republic

Yuh-Jzer Joung, National Taiwan University,
Taiwan

Carlos Juiz, Universitat de les Illes Balears, Spain

Katerina Kabassi, Tei of the Ionian Islands, Greece

Georgia Kapitsaki, University of Cyprus, Cyprus

George Karabatis, Umc, U.S.A.

Sokratis Katsikas, University of Piraeus, Greece

Natalya Keberle, Zaporozhye National University,
Ukraine

Matthias Klusch, Deutsches Forschungszentrum
für Künstliche Intelligenz, Germany

In-Young Ko, Korea Advanced Institute of Science
and Technology, Korea, Republic of

Waldemar W. Koczkodaj, Laurentian University,
Canada

Hiroshi Koide, Kyushu Institute of Technology,
Japan

Fotis Kokkoras, TEI of Thessaly, Greece

Agnes Koschmider, KIT - U, Germany

Karl-Heinz Krempels, RWTH Aachen University,
Germany

Tsvi Kuflik, The University of Haifa, Israel

Peep Kungas, University of Tartu, Estonia

Daniel Lemire, TELUQ, Canada

Stefania Leone, ETH Zürich, U.S.A.

Kin Fun Li, University of Victoria, Canada

Weigang Li, University of Brasilia, Brazil

Xitong Li, MIT Sloan School of Management,
U.S.A.

Dongxi Liu, CSIRO, Australia

Xiaozhong Liu, Indiana University, U.S.A.

Xumin Liu, Rochester Institute of Technology,
U.S.A.

Ying Liu, Cardiff University, U.K.

PROGRAM COMMITTEE (CONT.)

Leszek Maciaszek, Wroclaw University of Economics, Poland and Macquarie University, Sydney, Australia

Cristiano Maciel, Universidade Federal de Mato Grosso, Brazil

Michael Mackay, Liverpool John Moores University, U.K.

Tim A. Majchrzak, University of Münster, Germany

Dwight Makaroff, University of Saskatchewan, Canada

Massimo Marchiori, University of Padua, Italy

Kazutaka Maruyama, Meisei University, Japan

Michael Melliar-Smith, University of California, U.S.A.

Tarek Melliti, University of Evry, France

Ingo Melzer, Daimler AG, Germany

Emilia Mendes, Blekinge Institute of Technology, Sweden

Panagiotis Metaxas, Wellesley College, U.S.A.

Abdelkrim Meziane, CERIST Alger, Algeria

Tommi Mikkonen, Institute of Software Systems, Tampere University of Technology, Finland

Valérie Monfort, Université de Paris I Panthéon Sorbonne, France

Louise Moser, University of California, Santa Barbara, U.S.A.

Tomasz Muldner, Acadia University, Canada

Stavros Nikolopoulos, University of Ioannina, Greece

Vit Novacek, Digital Enterprise Research Institute, Nuig, Ireland

Jeff Z. Pan, University of Aberdeen, U.K.

Laura Papaleo, Province of Genova, Italy

Kyparisia Papanikolaou, ASPETE, Greece

Marcin Paprzycki, Polish Academy of Sciences, Poland

Eric Pardede, La Trobe University, Australia

Kalpdram Passi, Laurentian University, Canada

Viviana Patti, University of Torino, Italy

David Paul, The University of Newcastle, Australia

Toon De Pessemier, Ghent University - iMinds, Belgium

Alfonso Pierantonio, University of L'Aquila, Italy

Luis Ferreira Pires, University of Twente, The Netherlands

Pierluigi Plebani, Politecnico Di Milano, Italy

Jim Prentzas, Democritus University of Thrace, Greece

Birgit Pröll, Johannes Kepler University Linz, Austria

Dana Al Qudah, University of Warwick, U.K.

Werner Retschitzegger, Johannes Kepler University, Austria

Thomas Risse, L3S Research Center, Germany

Thomas Ritz, FH Aachen, Germany

Davide Rossi, University of Bologna, Italy

Gustavo Rossi, Lifa, Argentina

Davide Di Ruscio, University of L'Aquila, Italy

Maytham Safar, Kuwait University, Kuwait

Aziz Salah, Université du Québec à Montréal, Canada

Yacine Sam, University Tours, France

Comai Sara, Politecnico di Milano, Italy

Anthony Savidis, Institute of Computer Science, FORTH, Greece

Bernhard Schandl, Gnows.com, Austria

Claudio Schifanella, Università degli Studi di Torino, Italy

Harald Schöning, Software AG, Germany

Hamida Seba, University Lyon 1, France

Jochen Seitz, Technische Universität Ilmenau, Germany

Marianna Sigala, University of the Aegean, Greece

PROGRAM COMMITTEE (CONT.)

Marten van Sinderen, University of Twente, The Netherlands

Richard Soley, Object Management Group, Inc., U.S.A.

Anna Stavrianou, Xerox Research Centre Europe, France

Nenad Stojanovic, FZI at the University of Karlsruhe, Germany

Hussein Suleman, University of Cape Town, South Africa

Christoph Terwelp, RWTH Aachen University, Germany

Dirk Thissen, RWTH Aachen University, Germany

Thanassis Tiropanis, University of Southampton, U.K.

Giovanni Toffetti, IBM Research Lab Haifa, Israel

Riccardo Torlone, Università Roma Tre, Italy

Guy Tremblay, Université du Québec à Montréal, Canada

Th. Tsiatsos, Department of Informatics, Aristotle University of Thessaloniki, Greece

George Tsihrintzis, University of Piraeus, Greece

Athina Vakali, Aristotle University, Greece

Geert Vanderhulst, Alcatel-Lucent Bell Labs, Belgium

Jari Veijalainen, University of Jyväskylä, Finland

Maria Esther Vidal, Universidad Simon Bolivar, Venezuela

Maria Virvou, University of Piraeus, Greece

Petri Vuorimaa, Aalto University, Finland

Mohd Helmy Abd Wahab, Universiti Tun Hussein Onn Malaysia, Malaysia

Fan Wang, Microsoft, U.S.A.

Jason Whalley, Strathclyde University, U.K.

Bebo White, Stanford University, U.S.A.

Maarten Wijnants, Hasselt University, Belgium

Manuel Wimmer, Technische Universität Wien, Austria

Marco Winckler, University Paul Sabatier (Toulouse 3), France

Viacheslav Wolfengagen, Institute JurInfoR, Russian Federation

Bin Xu, Tsinghua University, China

Guandong Xu, University of Technology Sydney, Australia

Amal Zouaq, Royal Military College of Canada, Canada

AUXILIARY REVIEWERS

Alberto De La Rosa Algarin, University of Connecticut, U.S.A.

Markus C. Beutel, RWTH Aachen University, Germany

Matteo Ciman, University of Padua, Italy

José Cordeiro, Polytechnic Institute of Setúbal / INSTICC, Portugal

Javier Espinosa, LAFMIA lab, France

Golnoosh Farnadi, Ghent University, The Netherlands

Sevket Gökay, RWTH Aachen University, Germany

Beatriz Gomez, University of the Balearic Islands, Spain

Nuno Pina Gonçalves, EST-Setúbal / IPS, Portugal

Wolfgang Kluth, RWTH Aachen University, Germany

Fangfang Li, AAI, Australia

José António Sena Pereira, IPS - ESTSetúbal, Portugal

Laura Po, University of Modena and Reggio Emilia, Italy

Michael Rogger, STI, Austria

Christian Samsel, RWTH Aachen University, Germany

Alexander Semenov, University of Jyväskylä, NRU ITMO, Finland

Xin Wang, University of Southampton, U.K.

Stefan Wueller, RWTH Aachen, Germany

SPECIAL SESSION PROGRAM COMMITTEE

SPECIAL SESSION ON BUSINESS APPS

Henning Heitkötter, University of Münster,
Germany

Adrian Holzer, Ecole Polytechnique Fédérale de
Lausanne (EPFL), Switzerland

Tim A. Majchrzak, University of Münster,
Germany

Laura Po, University of Modena and Reggio
Emilia, Italy

Mohammad Tafiqur Rahman, Royal Institute of
Technology (KTH), Sweden

Davide Rossi, University of Bologna, Italy

Johannes Schobel, Ulm University, Germany

Virpi Kristiina Tuunainen, Aalto University
School of Business, Finland

SELECTED PAPERS BOOK

A number of selected papers presented at WEBIST 2014 will be published by Springer-Verlag in a LNBIP Series book. This selection will be done by the Conference Chair and Program Chair, among the papers actually presented at the conference, based on a rigorous review by the WEBIST 2014 Program Committee members.

FOREWORD

This book contains the proceedings of the 10th International Conference on Web Information Systems and Technologies (WEBIST 2014) which was organized and sponsored by the Institute for Systems and Technologies of Information, Control and Communication (INS-TICC). The conference was technically sponsored by the European Research Center for Information System (ERCIS).

The purpose of this Conference was to bring together researchers, engineers and practitioners interested in technological advances and business applications of web-based information systems. WEBIST had five main topic areas, covering different aspects of Web Information Systems, including “Internet Technology”, “Web Interfaces and Applications”, “Society, e-Business, e-Government”, “Web Intelligence” and “Mobile Information Systems”. We believe the proceedings here published demonstrate new and innovative solutions, and highlight technical problems in each field that are challenging and worthwhile.

The conference was also complemented with one special session, namely the Special Session on Business Apps - BA 2014 (chaired by Tim A. Majchrzak).

WEBIST 2014 received 153 paper submissions from 49 countries in all continents. A double-blind review process was enforced, with the help of 214 experts from the international program committee, all of them with a Ph.D. in one of the main conference topic areas. From these paper submissions only 23 papers were selected to be published and presented as full papers, i.e. completed work (12 pages in proceedings / 30’ oral presentations) and 41 additional papers, describing work-in-progress as short papers for 20’ oral presentation. Furthermore 35 short papers will be also presented as posters. The full-paper acceptance ratio was 15,03%, and the total oral paper acceptance ratio was 41,83%. These ratios denote a high level of quality, which we intend to maintain or reinforce in the next edition of this conference.

The high quality of the WEBIST 2014 programme is enhanced by four keynote lectures, delivered by experts in their fields, including (alphabetically): Andreas Pfeiffer (Hubject GmbH - joint venture of BMW Group, Bosch, Daimler, EnBW, RWE & Siemens, Germany), Fabien Gandon (INRIA, France), Steven Willmott (3scale, Spain) and Zakaria Maamar (Zayed University, United Arab Emirates).

Besides the proceedings edited by SCITEPRESS, a post-conference book will be compiled with extended versions of the conference’s best papers, and published by Springer-Verlag. Appropriate indexing has been arranged for the proceedings of WEBIST 2014 including Thomson Reuters Conference Proceedings Citation Index, INSPEC, DBLP, EI and Scopus. Furthermore, all presented papers will be available at the SCITEPRESS digital library.

The best contributions to the conference and the best student submissions were distinguished with awards based on the best combined marks of paper reviewing, as assessed by the

Program Committee, and the quality of the presentation, as assessed by session chairs at the conference venue.

In conjunction with a set of related conferences, namely SMARTGREENS and CLOSER, WEBIST had a new kind of session, named The European Project Space (EPS). The EPS aimed to provide insights into the research projects that are currently going on in Europe within the fields of web-based information systems, Cloud Computing and Smart Grids and Green IT Systems. This initiative intended to facilitate opportunities for knowledge and technology sharing, and establish the basis for future collaboration networks involving current project partners and interested conference delegates.

Building an interesting and successful program for the conference required the dedicated effort of many people. Firstly, we must thank the authors, whose research and development efforts are recorded here. Secondly, we thank the members of the program committee and additional reviewers for their diligence and expert reviewing. We also wish to include here a word of appreciation for the excellent organization provided by the conference secretariat, from INSTICC, which has smoothly and efficiently prepared the most appropriate environment for a productive meeting and scientific networking. Last but not least, we thank the invited speakers for their invaluable contribution and for taking the time to synthesize and deliver their talks.

Valérie Monfort

Université de Paris1 Panthéon Sorbonne, France

Karl-Heinz Krempels

RWTH Aachen University, Germany

CONTENTS

INVITED SPEAKERS

KEYNOTE SPEAKERS

- Software is Eating the World, APIs are Eating Software IS-5
Steven Willmott
- Typed Graphs and Linked Data - Modelling and Analyzing Social-semantic Web Data IS-7
Fabien Gandon
- Accelerating the Electric Mobility Market Through New Services and an Efficient Market Mode IS-11
Andreas Pfeiffer
- Enterprise 2.0 - Research Challenges and Opportunities IS-13
Zakaria Maamar

WEB INTERFACES AND APPLICATIONS

FULL PAPERS

- The One Hand Wonder - A Framework for Enhancing One-handed Website Operation on Touchscreen Smartphones 5
Karsten Seipp and Kate Devlin
- Contextinator - Project-based Management of Personal Information on the Web 14
Ankit Ahuja, Ben Hanrahan and Manuel A. Pérez-Quñones

SHORT PAPERS

- SafeMash - A Platform for Safety Mashup Composition 27
Carlo Marcelo Revoredo da Silva, Ricardo Batista Rodrigues, Rafael Roque de Souza and Vinicius Cardoso Garcia
- GeoSPARQL Query Tool - A Geospatial Semantic Web Visual Query Tool 33
Ralph Grove, James Wilson, Dave Kolas and Nancy Wiegand
- SIWAM: Using Social Data to Semantically Assess the Difficulties in Mountain Activities 41
Javier Rincón Borobia, Carlos Bobed, Angel Luis Garrido and Eduardo Mena
- Using Healthcare Planning Features to Drive Scientific Workflows on the Web 49
Bruno S. C. M. Vilar, André Santanchè and Claudia Bauzer Medeiros
- Linked Data Strategy to Achieve Interoperability in Higher Education 57
Guillermo García Juanes, Alioth Rodríguez Barrios, José Luis Roda García, Laura Gutiérrez Medina, Rita Díaz Adán and Pedro González Yanes
- Interdependent Components for the Development of Accessible XUL Applications for Screen Reader Users 65
Xabier Valencia, Myriam Arrue, Halena Rojas-Valduciel and Lourdes Moreno
- Hypermodal - Dynamic Media Synchronization and Coordination between WebRTC Browsers 74
Li Li, Wen Chen, Zhe Wang and Wu Chou

| | |
|--|-----|
| Integrating Adaptation and HCI Concepts to Support Usability in User Interfaces - A Rule-based Approach <i>Luisa Fernanda Barrera, Angela Carrillo-Ramos, Leonardo Florez-Valencia, Jaime Pavlich-Mariscal and Nadia Alejandra Mejia-Molina</i> | 82 |
| Tactive, a Framework for Cross Platform Development of Tabletop Applications <i>Ombretta Gaggi and Marco Regazzo</i> | 91 |
| On Metrics for Measuring Fragmentation of Federation over SPARQL Endpoints <i>Nur Aini Rakhmawati, Marcel Karnstedt, Michael Hausenblas and Stefan Decker</i> | 99 |
| Development Process and Evaluation Methods for Adaptive Hypermedia <i>Martin Balík and Ivan Jelínek</i> | 107 |
| CAPTCHA and Accessibility - Is This the Best We Can Do? <i>Lourdes Moreno, María González and Paloma Martínez</i> | 115 |
| Fuzzy-Ontology-Enrichment-based Framework for Semantic Search <i>Hajer Baazaoui-Zghal and Henda Ben Ghezala</i> | 123 |
| A Semantic-based Data Service for Oil and Gas Engineering <i>Lina Jia, Changjun Hu, Yang Li, Xin Liu, Xin Cheng, Jianjun Zhang and Junfeng Shi</i> | 131 |
| Cloud Space - Web-based Smart Space with Management UI <i>Anna-Liisa Mattila, Kari Systä, Jari-Pekka Voutilainen and Tommi Mikkonen</i> | 137 |
| Sequential Model of User Browsing on Websites - Three Activities Defined: Scanning, Interaction and Reading <i>Aneta Bartuskova and Ondrej Krejcar</i> | 143 |
| Automated Usability Testing for Mobile Applications <i>Wolfgang Kluth, Karl-Heinz Krempels and Christian Samsel</i> | 149 |

WEB INTELLIGENCE

FULL PAPERS

| | |
|--|-----|
| The GENIE Project - A Semantic Pipeline for Automatic Document Categorisation <i>Angel L. Garrido, Maria G. Buey, Sandra Escudero, Alvaro Peiro, Sergio Ilarri and Eduardo Mena</i> | 161 |
| Comparing Topic Models for a Movie Recommendation System <i>Sonia Bergamaschi, Laura Po and Serena Sorrentino</i> | 172 |
| Product Feature Taxonomy Learning based on User Reviews <i>Nan Tian, Yue Xu, Yuefeng Li, Ahmad Abdel-Hafez and Audun Josang</i> | 184 |
| Automatic Web Page Classification Using Visual Content <i>António Videira and Nuno Goncalves</i> | 193 |
| User Semantic Model for Dependent Attributes to Enhance Collaborative Filtering <i>Sonia Ben Ticha, Azim Roussanly, Anne Boyer and Khaled Bsaïes</i> | 205 |

SHORT PAPERS

| | |
|--|-----|
| Extracting Multi-item Sequential Patterns by Wap-tree Based Approach <i>Kezban Dilek Onal and Pinar Karagoz</i> | 215 |
|--|-----|

| | |
|--|-----|
| Improving Opinion-based Entity Ranking <i>Christos Makris and Panagiotis Panagopoulos</i> | 223 |
| Handling Weighted Sequences Employing Inverted Files and Suffix Trees <i>Klev Diamanti, Andreas Kanavos, Christos Makris and Thodoris Tokis</i> | 231 |
| XML Approximate Semantic Query based on Ontology <i>Yunkai Zhu, Chunhong Zhang and Yang Ji</i> | 239 |
| Finding Domain Experts in Microblogs <i>Shao Xianlei, Zhang Chunhong and Ji Yang</i> | 247 |
| Comparison between LSA-LDA-Lexical Chains <i>Costin Chiru, Traian Rebedea and Silvia Ciotec</i> | 255 |
| Prediction of Human Personality Traits From Annotation Activities <i>Nizar Omheni, Omar Mazhoud, Anis Kalboussi and Ahmed HadjKacem</i> | 263 |
| Towards Automatic Building of Learning Pathways <i>Patrick Siehndel, Ricardo Kawase, Bernardo Pereira Nunes and Eelco Herder</i> | 270 |
| A Survey on Challenges and Methods in News Recommendation <i>Özlem Özgöbek, Jon Atle Gulla and R. Cenk Erdur</i> | 278 |
| Combining Learning-to-Rank with Clustering <i>Efstathios Lempesis and Christos Makris</i> | 286 |
| Automated Identification of Web Queries using Search Type Patterns <i>Alaa Mohasseb, Maged El-Sayed and Khaled Mahar</i> | 295 |
| A Domain Independent Double Layered Approach to Keyphrase Generation <i>Dario De Nart and Carlo Tasso</i> | 305 |
| A Methodology to Measure the Semantic Similarity between Words based on the Formal Concept Analysis <i>Yewon Jeong, Yiyeon Yoon, Dongkyu Jeon, Youngsang Cho and Wooju Kim</i> | 313 |
| A Recommendation System for Specifying and Achieving S.M.A.R.T. Goals <i>Romain Bardiau, Magali Seguran, Aline Senart and Ana Maria Tuta Osman</i> | 322 |
| An Self-configuration Architecture for Web-API of Internet of Things <i>Eric Bernardes Chagas Barros and Admilson de Ribamar L. Ribeiro</i> | 328 |
| A Comparison of Three Pre-processing Methods for Improving Main Content Extraction from Hyperlink Rich Web Documents <i>Moheb Ghorbani, Hadi Mohammadzadeh and Abdolreza Nazemi</i> | 335 |
| Detection of Semantic Relationships between Terms with a New Statistical Method <i>Nesrine Ksentini, Mohamed Tmar and Faïez Gargouri</i> | 340 |
| An Approach to Detect Polarity Variation Rules for Sentiment Analysis <i>Pierluca Sangiorgi, Agnese Augello and Giovanni Pilato</i> | 344 |
| A General Evaluation Framework for Adaptive Focused Crawlers <i>Fabio Gasparetti, Alessandro Micarelli and Giuseppe Sansonetti</i> | 350 |
| A Domotic Ecosystem Driven by a Networked Intelligence <i>Luca Ferrari, Matteo Gioia, Gian Luca Galliani and Bruno Apolloni</i> | 359 |

SPECIAL SESSION ON BUSINESS APPS

FULL PAPERS

Towards Process-driven Mobile Data Collection Applications - Requirements, Challenges, Lessons Learned 371

Johannes Schobel, Marc Schickler, Rüdiger Pryss, Fabian Maier and Manfred Reichert

Location-based Mobile Augmented Reality Applications - Challenges, Examples, Lessons Learned 383

Philip Geiger, Marc Schickler, Rüdiger Pryss, Johannes Schobel and Manfred Reichert

SHORT PAPER

Alternative Communication System for Emergency Situations 397

I. Santos-González, A. Rivero-García, P. Caballero-Gil and C. Hernández-Goya

AUTHOR INDEX 403

INVITED SPEAKERS

KEYNOTE SPEAKERS

Software is Eating the World, APIs are Eating Software

Steven Willmott

3scale, Spain

Abstract: Software is becoming increasingly critical in many industries - from optimizing processes and information sharing, to powerful data analysis and embedded controllers. At the same time however, this software is becoming ever more componentized and inter-connected. Using core web technologies such as HTTP, XML, JSON and others, Web APIs are becoming the glue that weaves together these multiplying software components. This talk covers how software can often dramatic and change the fortunes of companies in a sector as well as how APIs and related Web Information technologies are part of this big transition.

BRIEF BIOGRAPHY

Steven was previously the research director of one of the leading European research groups in Europe on distributed systems and Artificial Intelligence at the Universitat Politècnica de Catalunya in Barcelona, Spain. He brings 10 years of technical experience in Web Services, Semantic Web, network technology and the management of large-scale international R&D teams.

Typed Graphs and Linked Data

Modelling and Analyzing Social-semantic Web Data

Fabien Gandon

INRIA, Univ. Nice Sophia Antipolis, CNRS, I3s, UMR 7271, 06900 Sophia Antipolis, France
Fabien.Gandon@Inria.Fr

Keywords: Semantic Web, Social Web, Typed Graphs, Knowledge Representation, Ontologies.

Abstract: This extended abstract summarizes the subject of an invited talk at WebIST 2014 that introduces methods, models and algorithms studied by the Wimmics research lab to bridge formal semantics and social semantics on the Web.

1 HYBRID WEB COMMUNITIES

The Web is now a virtual place where users and software interact in hybrid communities. These large scale interactions create many problems in particular the one of reconciling formal semantics of computer science (e.g. logics, ontologies, typing systems, etc.) on which the Web architecture is built, with soft semantics of people (e.g. posts, tags, status, etc.) on which the Web content is built.

The Wimmics research lab studies methods, models and algorithms to bridge formal semantics and social semantics on the Web. The name of the lab stands for “web-instrumented man-machine interactions, communities, and semantics” and we address this problem focusing on the characterization of typed graphs formalisms to model and capture these different pieces of knowledge and hybrid operators to process them jointly.

The approaches we introduce rely on graph-oriented knowledge representation, reasoning and operationalization to model and support actors, actions and interactions in web-based epistemic communities. The research results are applied to support and foster interactions in online communities and manage their resources.

This extended abstract introduces: the need to combine different kinds of graph-based representations (section 2), the importance of having a multidisciplinary approach (section 3); and the interest of relying on typed-graph knowledge representations. The last section provides a number of references.

2 FORMAL-SOCIAL SEMANTICS

The initial graph of linked pages of the Web has been joined by a growing number of other graphs: social networks, workflows, navigation trails, automatas of distributed services, linked open data clouds, etc. And these graphs interact in all sorts of ways influencing the life-cycle of each other.

Not only do we need means to represent and analyse each kind of graphs, we also need the means to combine them and to perform multi-criteria analysis on their combination.

We propose to address this problem by focusing on the characterization of (a) typed graphs formalisms to model and capture these different pieces of knowledge with their links and (b) hybrid operators to process them jointly. We especially consider the problems that occur in such structures when we blend formal stable semantic models and socially emergent and evolving semantics.

3 MULTIDISCIPLINARY MODEL

Our research follows two main directions combining two complementary types of contributions:

- First research direction: to propose multidisciplinary approach to analyse and model the many aspects of these intertwined information systems, their communities of users and their interactions;
- Second research direction: to propose formalizations of the previous models and reasoning algorithms on these models

providing new analysis tools and indicators, and supporting new functionalities and better management.

In a nutshell, the first research direction looks at models of systems, users, communities and interactions while the second research direction considers formalisms and algorithms to represent them and reason on their representations.

In the short term we intend to survey, extend, formalize and provide reasoning means over models representing systems, resources, users and social links in the context of social semantic web applications.

In the longer term we intend to extend these models (e.g. dynamic aspects), unify their formalisms (dynamic typed graphs) and propose mixed operations (e.g. metrical and logical reasoning) and algorithms to scale them (e.g. random walks) to support the analysis of epistemic communities' structures, resources and dynamics.

Ultimately our goal is to provide better collective applications on the web of data and the semantic web with two sides to the problem: (1) improve access and use of the linked data for epistemic communities and at the same time (2) use typed graph formalisms to represent the web resources, users and communities and reason on them to support their management.

4 TYPED GRAPHS

The work presented in this invited talk reports on models and algorithms to bridge formal semantics and social semantics by formalizing and reasoning heterogeneous semantic graphs.

The models we design include: users, their profiles, their requirements, their activities and their contexts; social links, social structures, social exchanges and processes; conceptual models including ontologies, thesauri, and folksonomies. Whenever possible these models are formalized and published according to standardized web formalisms and may motivate research and suggestions on extending these standards. The main goal here is to improve the understanding the systems have of the communities, resources and activities of their users. We focus on formalizing the models as unified typed graphs in order for software to exploit them jointly in their processing while still being able to use the specificities of each one of them. The schemas and datasets are published as linked data following the web architecture principles and W3C standards.

The algorithms we designed include: typed graphs indexing, reasoning and searching; hybrid processing merging logical inferences, rules and metrical inferences; approximation and propagation algorithms; distributed querying and reasoning. We propose algorithms (in particular graph-based reasoning) and approaches (in particular knowledge-based approaches) to process the mixed representations we obtain. We are especially interested in allowing cross-enrichment between them and in exploiting the life cycle and specificities of each one to foster the life-cycles of the others. These algorithms are published, implemented and distributed as part of a generic open source platform and library called Corese/KGram.

REFERENCES

- Wimmics web site, 2014, <http://wimmics.inria.fr/>
- Marie, N., Ribiere, M., Gandon, F., Rodio, F., 2013. *Discovery Hub: on-the-fly linked data exploratory search*, I-Semantics.
- Villata, S., Costabello, L., Gandon, F., Faron-Zucker, C., Buffa, M., 2013, *Social Semantic Network-based Access Control*, Security and Privacy Preserving in Social Networks, Lecture Notes in Social Networks, Springer
- Cojan, J., Cabrio, E., Gandon, F., 2013. *Filling the Gaps Among DBpedia Multilingual Chapters for Question Answering*, ACM Web Science Conference
- Buffa, M., Ereteo, G., Limpens, F., Gandon, F., 2013. *Folksonomies and Social Network Analysis in a social Semantic Web*, 39th International Conf. on Current Trends in Theory and Practice of Computer Science.
- Corby, O., Gaignard, A., Faron-Zucker, C., Montagnat, J., 2012. *KGRAM Versatile Data Graphs Querying and Inference Engine*, Proc. IEEE/WIC/ACM International Conference on Web Intelligence
- Corby, O., Faron-Zucker, C., 2010. *The KGRAM Abstract Machine for Knowledge Graph Querying*, IEEE/WIC/ACM International Conference.
- Erétéo, G., Buffa, M., Gandon, F., Corby, O., 2009. *Analysis of a Real Online Social Network using Semantic Web Frameworks*. International Semantic Web Conference, ISWC'09.
- Monnin, A., Limpens, F., Gandon, F., Laniado, D., 2010. *Speech acts meets tagging: NiceTag ontology*, AIS SigPrag International Pragmatic Web Conference Track, I-Semantics.
- Basse, A., Gandon, F., Mirbel, I., Lo, M., 2010. *Frequent Graph Pattern to Advertise the Content of RDF Triple Stores on the Web*, Web Science Conference

BRIEF BIOGRAPHY

Fabien Gandon is Senior Research Scientist and HDR in Informatics and Computer Science at INRIA and he is the Leader of the Wimmics team at the Sophia-Antipolis Research Center. He is also a member of the World-Wide Web Consortium (W3C) where he participates in several standardization groups. His professional interests include: Web, Semantic Web, Social Web, Ontologies, Knowledge Engineering and Modelling, Mobility, Privacy, Context-Awareness, Semantic Social Network/ Semantic Analysis of Social Network, IntraWeb. He previously worked for the Mobile Commerce Laboratory of Carnegie Mellon in Pittsburgh.

Accelerating the Electric Mobility Market Through New Services and an Efficient Market Mode

Andreas Pfeiffer

Hubject GmbH - joint venture of BMW Group, Bosch, Daimler, EnBW, RWE & Siemens, Germany

Abstract: How to connect different providers of electric mobility services and ensure an efficient design? An eRoaming platform as well as the contractual and technical framework required to implement this concept throughout Europe have already been developed by the Berlin-based joint venture Hubject. In addition an open interface protocol, the “Open InterCharge Protocol” (OICP), has been defined and can be accessed online by market participants for free since March 2013. Hubject GmbH is a joint venture formed by the BMW Group, Bosch, Daimler, EnBW, RWE and Siemens based in Berlin. The company was formed in 2012 and operates a cross-industry business and IT platform connecting infrastructure, service and mobility providers throughout Europe.

BRIEF BIOGRAPHY

Andreas Pfeiffer studied Business Administration at the RWTH Aachen University after training to become an information technology consultant. Upon successful completion of the degree, he worked as an organisational developer for an IT service provider. In 2007, he moved to Energieversorgungs- und Verkehrsgesellschaft mbH Aachen (E.V.A.) where he worked as a corporate developer. He managed the topic “electric mobility” for the Energieversorgungs- und Verkehrsgesellschaft mbH Aachen (E.V.A.) from 2008 to 2011 and was Managing Director of smartlab Innovationsgesellschaft mbH in Aachen from 2010 to 2012. Andreas Pfeiffer is Managing Director of Hubject GmbH, a joint venture of BMW Group, Bosch, Daimler, EnBW, RWE and Siemens based in Berlin, since August 2012.

Enterprise 2.0

Research Challenges and Opportunities

Zakaria Maamar

Zayed University, Dubai, United Arab Emirates

EXTENDED ABSTRACT

Today's enterprises are caught in the middle of a major financial crisis that is undermining their profit and growth and even jeopardizing their survival. To respond to this crisis, enterprises have launched different initiatives to improve their business processes and align their strategies with market needs, for example. Regular enterprise applications implement structured business processes in which the steps to perform are well defined. While this is beneficial for enterprises, there is little room for creativity in performing such processes without triggering a complex re-thinking process. Usually this process takes time to implement and analyze its effects, which is sometimes late and inefficient due to business constant changes. Several decisions are to be made on the fly and could tap into the large volume of data that Web 2.0 (social) applications (e.g., social networks, blogs, and wikis) generate.

Social applications rely on users' ability and willingness to interact, share, and recommend. However the richness and complexity of information in these applications pose challenges for enterprises on how to capture and structure this information for future use while preserving user privacy and information sensitivity. The social "fever" has caught every single activity of people's daily life ranging from sharing live experiences online to seeking feedback on any matter like what to wear for a special occasion. A report encourages enterprises to allow their employees to embrace social applications in order to establish and foster contacts with their colleagues, customers, and suppliers (B. Peter and R. Reeves. *Network Citizens: Power and Responsibility at Work*, Demos, 29 October 2008, www.demos.co.uk/publications/networkcitizens). This should impact in a positive way productivity, business development, and collegiality as long as these applications are used in a controlled way.

This presentation offers a research roadmap on the challenges and opportunities that Enterprise 2.0

(aka Social Enterprise) has to deal with and tap into, respectively. Contrary to traditional enterprises with a top-down command flow and bottom-up feedback flow, these flows in Enterprise 2.0 cross all levels and in all directions bringing all people together for the development of creative and innovative products and services. McAfee was the first to introduce the term Enterprise 2.0 as the use of emergent social software platforms within or between companies and their partners or customers (A. P. McAfee, *Enterprise 2.0: The Dawn of Emergent Collaboration*, MIT Sloan Management Review, 47(3), 2006). Enterprise social software should work hand-in-hand with regular business processes to ensure Enterprise 2.0 success (BLUEKIWI, *The State of Social Business 2013: The Maturing of Social Media into Social, Business*, blueKiwi report, 2013): "Enterprise 2.0 only works if it is part of a business process. It's great to work in new ways, but it's not enough. To make it real, it has to be very practical". Enterprise 2.0 should develop techniques and guidelines that would allow weaving social relationships (e.g., collegiality, fairness, and trustworthiness) into their operation. This should lead into a new form of business processes in the enterprise that reinforce the fact that employees establish and maintain social networks of contacts, rely on some privileged contacts when needed, and form with other peers strong and long lasting social collaborative groups. In today's economies, an enterprise's ability to sustain its growth and competitiveness depends on how well it manages from a social perspective its communications with stakeholders that are customers, suppliers, competitors, and partners (Y. Badr and Z. Maamar. *Can Enterprises Capitalize on Their Social Networks?* Cutter IT Journal, Special Issue on Measuring the Success of Social Networks in the Enterprise, 22(10), October 2009).

BRIEF BIOGRAPHY

Zakaria Maamar is a Professor and Assistant Dean in the College of Information Technology at Zayed University in Dubai, United Arab Emirates. His research interests are primarily related to service-oriented computing, social computing, and system interoperability. Dr. Maamar graduated for his M.Sc. and Ph.D. in Computer Sciences from Laval University in Canada in 1995 and 1998, respectively. Related Experiences: Organizer of several workshops in the past Quan Z. Sheng, Aikaterini Mitrokotsa, Sherali Zeadally, Zakaria Maamar (Eds.): RFID Technology-Concepts, Applications, Challenges, Proceedings of the 4th International Workshop, IWRT 2010, In conjunction with ICEIS 2010, Funchal, Madeira, Portugal, June 2010. SciTePress 2010, ISBN 978-989-8425-11-9 Soraya Kouadri Mostéfaoui, Zakaria Maamar, George M. Giaglis (Eds.): Ubiquitous Computing, Proceedings of the 3rd International Workshop on Ubiquitous Computing, IWUC 2006, In conjunction with ICEIS 2006, Paphos, Cyprus, May 2006. INSTICC Press 2006, ISBN 978-972-8865-51-1

WEB INTERFACES AND APPLICATIONS

FULL PAPERS

The One Hand Wonder

A Framework for Enhancing One-handed Website Operation on Touchscreen Smartphones

Karsten Seipp and Kate Devlin

*Department of Computing, Goldsmiths College, University of London, Lewisham Way, SE14 6NW, London, U.K.
{k.seipp, k.devlin}@gold.ac.uk*

Keywords: One-handed Operation, Mobile, Web, Wheel Menu, Interface Adaptation, CSS3, JavaScript, HTML.

Abstract: Operating a website with one hand on a touchscreen mobile phone remains a challenging task: solutions to adapt websites for mobile users do not address the ergonomic peculiarities of one-handed operation. We present the design and evaluation of the One Hand Wonder (OHW) – an easily-adaptable cross-platform JavaScript framework to support one-handed website navigation on touchscreen smartphones. It enhances usability without the need to redesign the existing website or to overwrite any CSS styles. User testing and quantitative evaluation confirm learnability and efficiency with clear advantages over non-enhanced browsing, and a discussion of the OHW’s versatility is given.

1 INTRODUCTION

The majority of smartphones sold today use modern operating systems such as Android and iOS, both of which have a powerful and largely standard-compliant browser paired with a touchscreen interface. Numerous websites already automatically adapt their layout and handling to the constraints of the access device to provide an adequate user experience, using web technologies such as JavaScript, CSS3 media queries or server-sided device detection.

With the help of established adaptation techniques for websites on mobile devices (W3C, 2008; ASA, 2007) as well as responsive themes (Envato, 2013), device-independent websites are becoming the norm. In addition, further approaches exist to adapt websites to mobile device constraints, although these are either proprietary (Akmin, 2012), dependent on a proxy server (Gupta et al., 2007) or bound to a specific browser (Mobotap, 2012; Yu and Miller, 2011). These result in an adapted and improved display on a range of mobile devices, but not in an adapted interaction model for thumb-based use.

For non-adapted pages, built-in actions such as pinching and tapping to zoom can improve matters. However, none of these account for the one-handed operation of the phone, which has been identified as a preferred mode of operation by many users (Karlson and Bederson, 2006). The limited mobility and reach of the thumb represent completely different

challenges to the designer regarding the layout and operation of the website; simply crafting it to ensure a correct display of the page elements does not suffice.

While some browsers (Mobotap, 2012) offer improvements for one-handed operation as part of their interface (using simple gestures, for example), we explore whether one-handed operation can be reliably improved regardless of the browser or plugin used. We do this by improving the display and control of elements operated via direct touch, without the need for gestures. As devices and browsers become increasingly powerful, the question arises as to whether such improvements can be made directly at runtime in the browser, and how efficient and usable these improvements are in comparison to non-enhanced websites. In this paper we present a JavaScript-based framework which we have named the One Hand Wonder (OHW). The OHW prototype provides an on-demand thumb-based interaction model for all interactive elements on a web page, facilitating operation and navigation across a wide range of websites. It gives quick access to the most common functions and elements and augments the interaction model of the browser and the standard HTML elements of a web page with additional one-handed UI features that can easily be toggled on and off. These augmentations are temporary and do not change the design of the website. The OHW is built using solely client-side technologies and is implemented by simply embedding the

code into the the web page. Initial user testing together with informal feedback during a demo session has confirmed acceptance and learnability. However, in this paper we assess more closely the usability, performance and practical implications of this approach and thus focus on:

1. Description of design and functionality of the framework and its interface.
2. Head-to-head comparison of the OHW's performance against normal, non-enhanced operation.
3. Discussion of its suitability for different types of websites based on its implementation into popular sites.
4. Overall discussion of the OHW as a tool for one-handed website operation on touchscreen smartphones.

2 PREVIOUS RESEARCH

When designing thumb-friendly interfaces, Wobbrock et al. (Wobbrock et al., 2008) suggest supporting and evoking horizontal thumb movements as much as possible, as vertical movements were found to be overly challenging. On this basis they suggest a horizontal layout of interactive elements on the screen to accommodate the ergonomic peculiarities of the thumb and improve usability. Katre (Katre, 2010) shows that a curved arrangement of elements on a touchscreen is perceived as comfortable and easy, as it supports a more natural circular motion of the thumb. An application of this is found in interfaces such as ArchMenu (Huot and Lecolinet, 2007) or ThumbMenu, where the user moves their thumb over an arch of elements placed in the bottom right corner of the screen.

Other researchers have explored the use of concentric menus such as the Wavelet menu (Francone et al., 2010) and SAM (Bonnet and Appert, 2011) to enhance thumb-based interaction, similar to the first generation Apple iPod. In the case of the Wavelet menu, research has shown that this approach with its consistent interaction model is easy to learn and efficient to use. Lü and Li (Lü and Li, 2011) present Gesture Avatar where the user can highlight a GUI element on screen to gain better control of it via an enlarged avatar. While this is an innovative way of improving one-handed device operation, it requires the user to draw their interface first, depends on a proprietary application and cannot be customised by the webmaster.

In terms of general adaptation of websites for mobile devices, one approach is web page segmenta-

tion (Hattori et al., 2007; Gupta et al., 2007), using a proxy server to re-render the page into new logical units which are subsequently served to the device. A proprietary solution is the Read4Me browser (Yu and Miller, 2011) where the browser offers to optimise the page for mobile display via a proxy-server that then serves it to the user. Bandelloni et al. suggest a different system (Bandelloni et al., 2005) where the developer creates an abstract XML-based description of the layout and a proxy server renders the information for the respective access device. In addition to these techniques, various services, themes and frameworks exist to adapt websites for mobile devices and CSS3 media queries offer a flexible approach for display adaptation.

While existing approaches are all concerned with the display of a website on mobile devices, the OHW addresses a so-far neglected aspect: the specific support of one-handed operation of the web page. By implementing the OHW into a website, improvements are made to the operation – rather than the presentation – of the site, as this remains problematic even on well-adapted sites when operating it with one hand. Most importantly, the enhancement is done at runtime and on the client, can be fully configured by the webmaster and is dependent only on the browser itself and the user, who can choose to switch the enhancements on or off at any time.

3 METHODOLOGY

To verify the findings of previous researchers (Katre, 2010; Wobbrock et al., 2008) promoting a curved interface for thumb-based GUIs, we conducted a small user study with 7 participants (3 F, mean age 31.43 years, SD 4.65), all of who declared to be frequent users of touchscreen mobile devices. Participants were asked to swipe 10 times using their right and left thumb without looking at the device. They were instructed to swipe in a way that was most natural and comfortable to them, avoiding bending and stretching of the joints. Traces of these swipes were recorded on a hidden layer and saved. Stacking the resulting images on top of each other shows the curved movement created by a horizontal swipe, supporting the findings of previous researchers and informing our design of the interface (Fig. 1). To develop and verify our design, we iteratively tested paper prototypes with users to transform the wheel menu metaphor into a comprehensive website interface, supporting a more natural operation and minimal strain. Building on discussions with web developers, we made the OHW as non-intrusive and supportive as possible in the form of

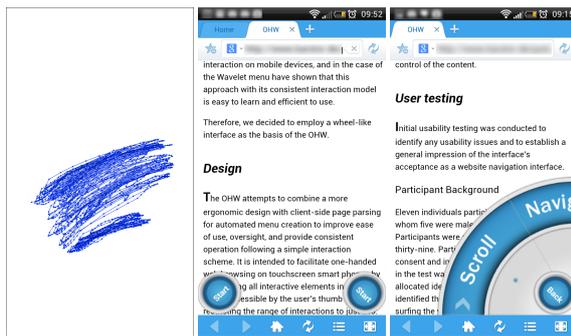


Figure 1: Visualisation of the swipe results (left), the OHW as it appears on start-up (middle) and launched (right).

an easily accessible, half-circle-shaped interface that can easily be added to a page by simply dropping the code into the website.

The OHW facilitates one-handed web browsing by assembling all interactive elements on request in a region easily accessible by the user's thumb and restricts the range of interactions to just two – swipe and tap. The layout of the page stays untouched and users can decide whether or not to use the interface at any time by switching it on or off (Fig. 1). Thus, the OHW is not an interface for mobile optimisation, which can be achieved using the techniques outlined above. Rather, the OHW's purpose is to enhance one-handed operation of a website regardless of its degree of adaptation, without spoiling the design. It augments the interaction model, not the display. To function, the OHW requires a browser with CSS3 support together with the jQuery JavaScript library – the most popular JavaScript library to date (Pingdom, 2010) – present on the website. Other than this, there are no minimum standards required and the OHW can be implemented into pages that already include libraries such as MooTools, for example. It has been trialled on a range of Android and iOS devices with HTML4 and HTML5 mark-up in Standards and Quirks mode.

The OHW interface consists of a variety of modules whose availability depends on the content of the website and the interface's configuration. Each module is represented as a wedge and together they form a wheel-type interface, either at the right-hand or left-hand bottom corner of the screen, depending upon the user's choice (Fig. 1). Only the modules that correspond to elements found on the page are loaded, but additional modules can be added at runtime by listening to updates of the Document Object Model (DOM).

To implement the OHW, the webmaster only needs to ensure that the jQuery JavaScript library is available on the website before linking to the OHW's code using a basic `<script>` tag. The webmaster

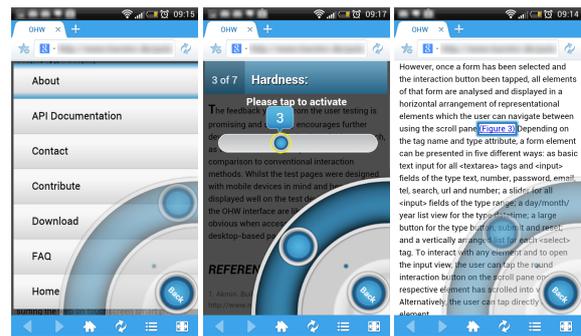


Figure 2: Basic list view of the site navigation (left), an augmented input field of the type range (middle) and the scroll functionality (right).

can optionally edit the configuration file, which is a JavaScript object, and adjust themes, selectors and custom functionality. As each and every aspect of the interface can be adjusted via CSS and HTML, the OHW can fit the look and content of a wide variety of websites. Once implemented, the code scans the website for certain tags from which to build the interface. By default these are basic HTML elements, such as `<nav>`, `<video>`, `<audio>`, `<form>`, `<h2>`, and `<a>`, and from these the standard names of the wedges are derived. This can easily be extended by using CSS selectors and custom wedges declared in the configuration file. The OHW contains several methods to cope with incorrect mark-up and can report any encountered problems to the webmaster.

The OHW's use is optional for the user and the interface can be hidden and brought back at any time. The interface is launched by tapping the Start button on either side of the screen to make it visible (Fig. 1). It can be spun by swiping over it to reveal all available functions. The Start button then becomes a Back button and can be used to either hide the interface completely or to go back one level. For example, if the user was standing and only had one hand free, they could tap the Start button and operate the site one-handedly with the help of the interface. As they sit down and free their other hand, they could hide the interface by tapping the Back button and continue to operate the website with both hands, without a change in design or presentation.

4 FUNCTIONALITY

The OHW offers improved presentation and one-handed operation for all interactive page elements. This can be achieved either by accessing them via the respective wedge in the wheel or by directly tapping them on the page. By default, the OHW uses basic



Figure 3: A checkbox input rendered by the OHW, where a tap on the OHW interface toggles the state (left), a user selecting a video from the media menu (middle) and video playback control (right).

list views (Fig. 2) that can hold images and text. In addition it offers on/off switches, sliders, buttons and a media player (Fig. 3, Fig. 4). These views are used in combination for the augmented presentations of otherwise hard to use elements and can be controlled by swiping and tapping. The OHW also provides scroll functionality similar to that of the Opera Mini browser (ASA, 2012). While scrolling, interactive elements closest to the current scroll position are outlined one at a time (Fig. 2) and can be activated by tapping the interface. Text input uses the system keyboard as initial tests showed that users found the OHW's own concentric keyboard hard to use. In addition to the functionality provided, the above can be easily extended by the webmaster by combining a CSS selector, one of the OHW's views and a custom function to suit their needs.

5 INITIAL USER TESTING

After iterative paper prototyping the interface was built and a pilot user test with 11 participants (6 F, all frequent smartphone users) was conducted to identify usability issues and to establish acceptance. Users were given a set of tasks one might perform on a page consisting of headlines, forms, videos, navigation and links, which they completed using the OHW without assistance. The page was designed to be device-independent using CSS3 media queries. All actions were recorded in video and audio, and feedback was given in a questionnaire on a five-point Likert scale. Feedback was predominantly positive and in response to the collected data we created an improved version of the interface for a usability study determining the efficiency and speed of the OHW in comparison to the normal operation (with one hand without the OHW) of a mobile-optimised website. During a demo ses-



Figure 4: OHW representations of form elements: A button (left), a text input field (middle) and a date field. Users can swipe over the interface to navigate between elements in a form and tap to engage with the active element. Each element can be operated by performing swipe and tap actions on the interface.

sion at a conference (Seipp and Devlin, 2013) we presented the improved interface to visitors, implementing it on a range of popular websites to be experienced “hands-on”. Informal feedback from users during these sessions was consistently positive, complimenting on the ease-of-use, usefulness and quality, thus supporting the validity of our approach to improve one-handed operation of websites on touchscreen mobile devices.

6 USABILITY STUDY

Altogether, 22 participants (7 F) aged 20 to 34 took part in the study, 19 of whom were final year undergraduate Computing students and the remainder were young professionals. All of them were right-handed, regular users of touchscreen mobile devices, such as phones and media players. The 19 Computing students were briefly introduced to the OHW during class and asked to do some self-directed exploring on a test page. On the day of the study, all users were given a 5-minute explanation of the usage of the OHW to ensure its operation was fully understood.

Using a within-subjects design, participants carried out the study one-handedly both with and without the use of the OHW. The study was counterbalanced by altering the mode in which the tasks were first performed. The first part of the usability study comprised 10 separate standard tasks a user might perform on a website and cover the whole spectrum of the OHW:

1. Finding a menu item in the navigation
2. Finding a video and forwarding it to a certain time
3. Finding a form and activating a checkbox
4. Finding another video and starting it

5. Finding a form and filling in a date
6. Finding a link in the body text
7. Finding a form and filling in a range value
8. Finding a headline
9. Finding a form and pressing a button
10. Scrolling and clicking on a link

The study featured a website presenting the OHW. It was coded in HTML5 and CSS3 and contained a page navigation (<nav>), headlines (<h1>, <h2>, <h3>), a form with various elements of input types (“text”, “range”, “datetime”, “submit”, “checkbox”), paragraphs of text (<p>), three video files with poster images (<video>), links (<a>), images () as well as various <div> elements for the layout. These elements are very common and can be considered as representative for many websites. CSS3 media queries and relative measures were used to make the website’s presentation device-independent. To conduct the study we used a HTC Sensation XE with Android 4.03 and the Maxthon Mobile Browser.

Tasks were performed directly on the website and were preceded by an instruction screen. Each task commenced from the top of the page. The number of interactions and time needed to accomplish the task were recorded using JavaScript. Recording began when users pressed OK on the instruction screen and stopped when a task was completed. For example, recording was only stopped once the target link was clicked or a certain value was entered into a field.

While the above is well-suited for determining efficiency on discrete tasks, it is less suitable for predicting the OHW’s “real-life” performance on a page containing any number of these elements, where spatial proximity could affect performance. To address this we also measured the performance in 10 additional, consecutive tasks (1c to 10c), mimicking a set of coherent actions. After each part of this use case, recording was paused to show the instructions for the next part, but the current state of the website (scroll position, opened menus etc.) remained unchanged:

- 1c. Navigating to a headline in the text
- 2c. Scrolling and clicking on a link
- 3c. Finding a menu item in the navigation
- 4c. Finding a video and forwarding it to a certain time
- 5c. Navigating to another headline in the text
- 6c. Finding a link in the body text by scrolling
- 7c. Finding a form and entering a word into a text field
- 8c. Activating a checkbox in the same form
- 9c. Filling in a date in the same form
- 10c. Pressing a button in the same form

7 RESULTS

The data was evaluated using a Wilcoxon signed-rank test. Due to the varying, skewed results and small sample size we chose a series of non-parametric tests over the ANOVA. As tasks were not comparable in their results because of their different nature, they had to be treated as separate. Comparison of the one-handed task performance of users with the OHW against the normal, non-enhanced way draws a clear picture of the benefits the OHW offers to one-handed website operation. The effect of the OHW on efficiency can be derived from the median number of interactions required to perform a task (Table 1) as well as from the time needed to complete it (Table 2). Note: The values of the use case (shown in the tables as C) are based on the time and interactions needed to complete the whole use case, consisting of the 10 additional parts 1c to 10c, forming one large task.

Table 1: Median interactions per task and the use case (C) w/out the OHW including Z and p values as well as % of interactions (I) needed with OHW (Normal = 100%).

| Task | OHW | Normal | Z | p | %I OHW |
|------|------|--------|------|--------|--------|
| 1 | 6 | 19 | 3.49 | < .001 | 32% |
| 2 | 14 | 13.5 | 0.63 | .526 | 104% |
| 3 | 14 | 22.5 | 2.93 | .003 | 62% |
| 4 | 9 | 12.5 | 2.1 | .036 | 72% |
| 5 | 26.5 | 27 | 0.15 | .884 | 98% |
| 6 | 7.5 | 16 | 3.9 | < .001 | 47% |
| 7 | 12 | 12.5 | 0.06 | .952 | 96% |
| 8 | 14 | 19 | 1.97 | .049 | 74% |
| 9 | 13.5 | 9.5 | 2.95 | .003 | 142% |
| 10 | 51.5 | 26 | 3.98 | < .001 | 198% |
| C | 104 | 127 | 2.18 | .029 | 82% |

7.1 Results: Number of Interactions Needed

In 5 out of 10 tasks, the OHW allows users to complete the task with fewer interactions (Table 1). This is most visible in Tasks 1, 3 and 6 where the same task could be accomplished with only 32%, 62% and 47% of the interactions required without the interface. This highlights the OHW’s enhancement of tasks such as finding an item in the navigation, operating a checkbox and retrieving a link from the body text. Other tasks that took fewer interactions to perform with the OHW than without were locating a video (Task 4, 72%) and finding a headline (Task 8, 74%). However, the results also show areas where more interac-

tions are required with the OHW than without. This includes finding and pressing a submit button (Task 9, 142%) and lengthy scrolling to find a link (Task 10, 198%).

7.2 Results: Amount of Time Needed

Evaluating OHW performance based on the actual time needed to complete a task draws an even clearer picture of the OHW's effectiveness (Table 2). Using the OHW in all tasks but one is significantly faster than performing the same tasks without the OHW. The most striking differences can be observed in Task 1 (33% of the time needed), Task 3 (47%), Task 4 (36%), Task 6 (33%) and Task 8 (46%). However, scrolling through the document (Task 10) takes more time with the OHW (147% of time needed).

Table 2: Median time (T) in seconds needed per task (1 to 10) and the use case (C) w/out the OHW including Z and p values as well as % of time needed with the OHW (Normal = 100%).

| Task | OHW | Normal | Z | p | %T OHW |
|------|--------|--------|------|-------|--------|
| 1 | 11.40 | 34.30 | 4.11 | <.001 | 33% |
| 2 | 25.90 | 45.10 | 4.07 | <.001 | 57% |
| 3 | 20.50 | 43.90 | 4.11 | <.001 | 47% |
| 4 | 10.20 | 28.60 | 4.11 | <.001 | 36% |
| 5 | 33.90 | 46.60 | 3.98 | <.001 | 73% |
| 6 | 9.50 | 29.10 | 4.11 | <.001 | 33% |
| 7 | 18.80 | 26.30 | 3.85 | <.001 | 72% |
| 8 | 14.80 | 31.90 | 4.07 | <.001 | 46% |
| 9 | 15.50 | 22.10 | 3.17 | .002 | 70% |
| 10 | 65.40 | 44.50 | 3.56 | <.001 | 147% |
| C | 153.20 | 255.10 | 4.11 | <.001 | 60% |

7.3 Results of the Use Case

The results of the use case show that in a real-life application the impact of the OHW on efficiency is significant, as overall it took participants only 60% of the time and 82% of the interactions when using the OHW as opposed to operating the website normally (Table 2 and Table 1).

7.4 Implementation into Popular Websites

To determine the OHW's versatility and suitability for different types of websites, we implemented it on several popular websites via a proxy script that injected

the OHW code into the loaded page with the following results:

7.4.1 Wikipedia

(Wikipedia, 2013) The mobile article view divides the content into sections which can be expanded, each headed by an <h2>. Out of the box the OHW was useful for quick access to the navigation, links, search and scrolling, but not as useful for jumping to a headline, as the user would still have to tap the element on the page to expand it. Overall the interface was quick to load and very responsive, but the number of links on the page shown in the OHW Links menu totalled 538. This resulted in a very large list with jerky scrolling despite the use of hardware-accelerated CSS transitions.

7.4.2 BBC News

(BBC, 2013) Standard configuration offered quick access to the page navigation and search form, but the headlines menu combined stories from all sections, making it hard for the user to discern where a story belonged. This indicates that the content to be made accessible by the OHW needs to be defined by the webmaster when configuring the OHW using custom wedges and selectors. Start-up was quick and operation was very smooth. The Headlines menu held 32 items, the Links menu 15 and the Navigation menu 35 items. Implementing the OHW on the desktop version of the site, however, exposes a weakness of this client-side approach. The desktop contains various JavaScript-based fading animations which heavily impact performance on mobile devices. With these animations, the otherwise smooth operation was occasionally interrupted as the OHW has to share the processing resources with other page elements.

7.4.3 W3C

(W3C, 2013) The OHW performed very smoothly with 20 items in the Links menu, 10 items in the Navigation menu and 22 items in the Headlines menu. As stated previously, the labels of the wedges could use simple customisation by the webmaster to reflect the content of the website.

7.4.4 Google

(Google, 2013) The OHW was quick to load and very responsive with 11 items in the Navigation menu, 10 items in the Results menu (Headlines), and 11 items in the Links menu. Some customisation is required to better match the content.

7.4.5 WordPress Blog

(WordPress, 2013) The OHW performed smoothly with 6 items in the Navigation menu, 16 items in the Links menu and 12 items in the Posts menu (Headlines), but again the wedges could be renamed to represent the content meaningfully.

7.4.6 YouTube

(YouTube, 2013) Smooth performance with a basic configuration showing 4 items in the Navigation menu, 9 items in the Videos (Headlines) menu and 6 items in the Links menu. Unfortunately, videos could not be played back within the OHW as these were served as 3gp files. The OHW can only play back files which are natively supported by the browser.

7.4.7 Flickr

We were unable to receive the mobile version of the site (Flickr, 2013) using our proxy script, despite manually altering the header information of the request to mimic a mobile device. Attempts to mirror parts of the site locally did not allow sufficiently accurate reconstruction of the mobile view either.

7.5 General Performance

Performance of the OHW (Table 3) was tested in the standard browser with varying amounts of content on an HTC Sensation XE with Android 4.03 and an iPhone 3GS with iOS 6.1. First, we measured the start-up time of the interface on each device on the websites discussed in the previous section. Then we measured the time it took each device to create a list view with varying amounts of items after tapping a wedge in the wheel. For this we chose the WordPress blog (WordPress, 2013) as a base and injected additional elements into the DOM when fetching the page using the proxy script. All measurements were performed three times on each device with a cleared browser cache.

8 DISCUSSION

First we discuss usability and efficiency from a user perspective. Next we discuss the performance of the framework to determine the boundaries in which it can be deployed.

Table 3: Mean time in seconds needed by the HTC Sensation XE (S. XE) and iPhone 3GS (3GS) to create a list view (Fig. 1, right) after tapping a wedge in the wheel. Second part shows mean start-up time (SU) when implemented on a website.

| Task | S. XE | 3GS |
|----------------------|-------|-----|
| List view, 30 items | 1.1 | 0.6 |
| List view, 60 items | 1.5 | 0.7 |
| List view, 120 items | 1.9 | 0.7 |
| List view, 480 items | 2.4 | 0.6 |
| SU Wikipedia | 1.4 | 0.9 |
| SU BBC News | 2.3 | 1.4 |
| SU W3C | 0.3 | 0.8 |
| SU Google | 0.3 | 1.0 |
| SU WordPress | 1.5 | 1.4 |
| SU YouTube | 0.9 | 0.9 |

8.1 Usability and Efficiency

When operating a website with only one hand, the OHW presents a clear advantage over the normal, non-enhanced mode of operation. In the majority of cases, using the OHW requires less interaction to complete a task and in 90% of the examined cases, a task is completed significantly faster when using the OHW. However, the results also highlight a weak point of the OHW. When the user has to scroll a large section of the page, the performance of the OHW is significantly weaker than the normal mode of operation (147% of time needed). It shows that scrolling with one hand is already very efficient and that the OHW's approach cannot compete with the existing solution, but needs improvement. This has since been addressed by combining native scrolling and OHW scrolling so that the user can scroll the website as usual, but can make more precise selections by moving their thumb over the interface at the same time.

8.2 Versatility

Implementation into different types of websites highlights the pros and cons of our approach. Customisation of the OHW is easy: the webmaster can quickly adapt the text of the wedges to reflect website content using the supplied templates. Custom functionality is achieved by using the OHW's plugin model to accommodate a website's own set of interactions, such as the accordion-like blocks on Wikipedia (Wikipedia, 2013) with custom callback functions. Thus configured, the OHW is suited to pages with categorised



Figure 5: The OHW on a Nexus 7 in horizontal orientation.

text and images that stretch over many screens and would otherwise need scrolling to access, as found on news websites, wikis, forums, blogs and search engines. Benefits for all types of websites include quick access and operation of forms, navigation, and control of audio or video items if supported natively by the browser.

The OHW performs well on mobile-adapted pages if the content per menu is not excessive. Operation is smooth even with 120 items to be displayed and scrolled. Beyond that the list scrolling performance decreases on the HTC Sensation XE with the amount of data to be presented, whereas it stays the same on the iPhone 3GS with up to 480 list items. While this decrease is likely to only happen in rare cases on mobile websites – as observed in our Wikipedia test – it is more likely to occur on desktop-oriented pages due to the larger page load and other resource-depleting processes. Therefore the webmaster has to be considerate when implementing the OHW: a site loaded with badly coded animations that already struggles being displayed on a mobile device will not necessarily be improved by the OHW. This highlights the main problem of using an integrated client-side approach: the interface has to share the resources with the content of the website, which can directly influence performance. Luckily, this is in the hands of the webmaster implementing the OHW and thus straightforward to address. However, it also shows that the OHW is not a magical one-size-fits-all solution for making any website easier to interact with when operating the device with one hand. What it does, though, is significantly improve operation and efficiency on already mobile-adapted websites (Table 2) together with a short start-up time and high responsiveness (Table 3).

9 CONCLUSION

Our research shows that applying the wheel-menu metaphor as the basis for thumb-based website interaction and offering a curved input control based on solely swipe and tap for all interactive elements clearly improves one-handed website operation and allows users to complete their goals more quickly and comfortably as they do not have to loosen their grip on the device when trying to reach elements outside the arc of their thumb. Given the demand for a simple, one-handed way to access websites on a mobile device (Karlson and Bederson, 2006), the OHW is a practical and highly effective solution from both a web developer and end-user perspective, if the technical requirements are met. The OHW promotes a free and inclusive way of improving user experience on the mobile web for modern touchscreen smartphone users and supports openness and flexibility. The use of standard web technologies allows it to easily adapt to new challenges and ensures its longevity and ease-of-use for the webmaster. Future work will address performance optimisations for the operation of large lists and the development of a SVG and core JavaScript implementation. We plan to evaluate the OHW's applicability as an interface for HTML5-based smartphone apps and the development of an extended plugin model to allow more advanced custom functionality. As it stands, the OHW is a promising approach for enhancing one-handed web browsing on a wide range of mobile touchscreen devices (Fig. 5).

REFERENCES

- Akmin (2012). Build your own mobile website ... in minutes. <http://www.mobisitegalore.com/index.html>.
- ASA, O. S. (2007). Making small devices look great. <http://dev.opera.com/articles/view/making-small-devices-look-great>.
- ASA, O. S. (2012). <http://www.opera.com/mobile/specs>.
- Bandelloni, R., Mori, G., and Paternò, F. (2005). Dynamic generation of web migratory interfaces. In *Proc. Mobile HCI 2005*, pages 83–90, New York, NY, USA. ACM.
- BBC (2013). BBC news. <http://m.bbc.co.uk/news>.
- Bonnet, D. and Appert, C. (2011). Sam: the swiss army menu. In *Proc. IHM 2011*, pages 5:1–5:4, New York, NY, USA. ACM.
- Envato (2013). Signum mobile — html5 & css3 and iwebapp. <http://theforest.net/item/signum-mobile-html5-css3-and-iwebapp/1614712>.
- Flickr (2013). Flickr. <http://m.flickr.com>.
- Francone, J., Bailly, G., Lecolinet, E., Mandran, N., and Nigay, L. (2010). Wavelet menus on handheld devices:

- stacking metaphor for novice mode and eyes-free selection for expert mode. In *Proc. AVI 2010*, AVI '10, pages 173–180, New York, NY, USA. ACM.
- Google (2013). Google search results. <https://www.google.co.uk/search?q=something>.
- Gupta, A., Kumar, A., Mayank, Tripathi, V. N., and Tapaswi, S. (2007). Mobile web: web manipulation for small displays using multi-level hierarchy page segmentation. In *Proc. MC07 4th Mobility Conference*, pages 599–606. ACM.
- Hattori, G., Hoashi, K., Matsumoto, K., and Sugaya, F. (2007). Robust web page segmentation for mobile terminal using content-distances and page layout information. In *Proc. WWW 2007*, pages 361–370. ACM.
- Huot, S. and Lecolinet, E. (2007). Archmenu et thumb-menu: contrôler son dispositif mobile "sur le pouce". In *Proc. IHM 2007*, pages 107–110, New York, NY, USA. ACM.
- Karlson, A. K. and Bederson, B. B. (2006). Studies in one-handed mobile design: Habit, desire and agility. Technical report, Computer Science Dept., Uni. of Maryland.
- Katre, D. (2010). One-handed thumb use on smart phones by semi-literate and illiterate users in india. In *HWID: Usability in Social, Cultural and Organizational Contexts*, volume 316, pages 189–208. Springer Boston.
- Lü, H. and Li, Y. (2011). Gesture avatar: a technique for operating mobile user interfaces using gestures. In *Proc. CHI 2011*, pages 207–216. ACM.
- Mobotap (2012). Dolphin browser. <http://dolphin-browser.com/>.
- Pingdom (2010). jQuerys triumphant march to success. <http://royal.pingdom.com/2010/03/26/jquery-triumphant-march-to-success/>.
- Seipp, K. and Devlin, K. (2013). Enhancing one-handed website operation on touchscreen mobile phones. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pages 3123–3126, New York, NY, USA. ACM.
- W3C (2008). Mobile web best practices 1.0. <http://www.w3.org/TR/mobile-bp/>.
- W3C (2013). W3C. <http://www.w3.org>.
- Wikipedia (2013). Deusdedit of Canterbury. http://en.m.wikipedia.org/wiki/Deusdedit_of_Canterbury.
- Wobbrock, J. O., Myers, B. A., and Aung, H. H. (2008). The performance of hand postures in front- and back-of-device interaction for mobile computing. *Int. J. Hum.-Comput. Stud.*, 66(12):857–875.
- WordPress (2013). Just another wordpress weblog. <http://en.blog.wordpress.com/>.
- YouTube (2013). Youtube mobile. <http://www.youtube.com/results?client=mv-google&q=sublime>.
- Yu, C.-H. and Miller, R. C. (2011). Enhancing mobile browsing and reading. In *Ext. Abstracts Proc. CHI 2011*, pages 1783–1788. ACM.

Contextinator - Project-based Management of Personal Information on the Web

Ankit Ahuja, Ben Hanrahan and Manuel A. Pérez-Quiñones

Department of Computer Science, Virginia Tech, Blacksburg, VA U.S.A.

{ahuja.ankit, hanrahan.ben}@gmail.com, perez@cs.vt.edu

Keywords: Information Fragmentation, Personal Information Management, Tool Integration, Web-based systems.

Abstract: The web browser is a central workspace for knowledge workers, where they use cloud-based applications to access their information. While this solution fits nicely within our diverse ecosystem of devices, it may reintroduce and proliferate faults of the desktop, particularly information fragmentation. Information fragmentation is an increasingly important issue on the cloud as information is typically silo-ed within different applications. This results in users replicating storage and organization due to the lack of a unifying structure. As cloud applications become more rich, the need to investigate whether these faults of the past are still problematic becomes more important. To probe this question we created *Contextinator*, a tool for the web browser that assists in coordinating data for projects. *Contextinator* enables knowledge workers to manage cloud-based information and project artifacts in a centralized place, providing a unifying structure. In this paper, we discuss the design of our system, and the results of our mixed-method evaluation. Our findings contribute insight into the need for, and appropriateness of, projects as unifying structures for the web. Our results point to two types of projects we call ‘preparatory’ and ‘opportunistic’ based on when and why users create them.

1 INTRODUCTION

The web browser has emerged as a central workspace for knowledge workers. This cloud-based approach complements our diverse ecosystem of devices as most of the application data resides in a device agnostic remote storage. This trend is further strengthened as mobile devices and web applications begin to match the functionality of desktop applications.

However, this remote storage comes at a cost, as data on the cloud is typically accessed through particular applications or services (e.g., Dropbox for files, Evernote for notes, Gmail for email). These applications create silos of data that are not always inter-operable with each other. These silos do not share a unifying structure and proliferate information fragmentation, a problem previously identified by the Personal Information Management (PIM) community (Karger and Jones, 2006). The siloing of data between different applications prevents the user from creating more salient groupings based on their real world relationships with the data.

Information fragmentation in this context has many undesired consequences. First, data is stored in separate applications and often only available through each particular application (e.g., an Evernote note-

book is not available in Gmail). Second, user-defined groupings are done in ad-hoc ways particularly if data for the group is scattered over different applications. For example, it is typical to find a Dropbox folder, a Gmail label/folder, and an Evernote notebook for a particular project but these three individual items are stored in different applications and there is no easy way for users to group them under a user-meaningful name, like “home repair project.” Instead users end up duplicating organizational hierarchies between tools (Boardman and Sasse, 2004; Boardman et al., 2003).

This cloud-based information fragmentation leads to another problem. Users work on the web having many windows or tabs open to access these online service. This in turn exacerbates the problem by requiring more management of windows, tabs, and bookmarks. Suspending and resuming work on a project, something we know is typical of today’s knowledge workers (Czerwinski et al., 2004), is made more difficult by these disconnected tabs and windows.

We built *Contextinator* as a way to study these problems. *Contextinator* is a tool that enables users to group their web sessions and cloud based artifacts into projects. *Contextinator* was built with compatibility in mind and is compatible with the majority of existing web-based tools.

In this paper, we present the related work in the areas of interest, ground our design decisions in previous research, present the results of our mixed-method evaluation, and close with implications for future systems. We not only find evidence that users need a way to group activities, but that the project metaphor is too restrictive. We have also gained insight into the complex way knowledge workers contextualize and think about their work.

2 RELATED WORK

Several areas of research are related to this work: Information Fragmentation, Task Management, Activity Based Computing, Multitasking, and Window Management. As such, we review the research for each of these areas and provide insight into how they shaped the design of *Contextinator*.

2.1 Information Fragmentation

Information fragmentation occurs when our personal information is scattered over different devices, storage systems, and online tools. Typically each of these have its own organizational structure and it is up to users to integrate information across these systems. Information fragmentation is considered a ‘pervasive problem in personal information management’ (Karger and Jones, 2006).

The siloing of information by applications is not a new problem unique to the cloud. In previous studies users were found to use several methods to create groupings in spite of these silos. These methods include: using multiple folder hierarchies to organize documents related to projects (Jones et al., 2005); using a special folder (or tag) in an email client to hold messages related to a project (González and Mark, 2004); or using virtual spaces to separate windows of different projects (Henderson and Card, 1986).

A drawback with this ad-hoc approach is that users end up maintaining duplicate organizational hierarchies between tools, which are difficult to maintain. Boardman et al. (Boardman and Sasse, 2004; Boardman et al., 2003) studied users’ PIM organization strategies across different tools and identified some of the problems caused by information fragmentation. Including: compartmentalization of data between distinct tools; difficulty in coordinating across different tools; and inconsistencies between equivalent functionality. To solve the second problem, Boardman et al. created a prototype to mirror folder structures between different PIM tools and

users found the sharing categories between tools intuitive and compelling.

Bergman et al. (Bergman et al., 2006) framed the problem as project fragmentation, where information was fragmented into different collections without relation to the common activity uniting them. Their solution was to use a single hierarchy to store all files of different formats under the same folder. Similarly, Jones et al. (Jones and Anderson, 2011) suggested the development of a common structure that could be shared and manipulated by any number of tools.

Integrating information collections in the cloud is also being pursued in the commercial and open source tool space. However, most of these tools assist users in accessing information, and do not endeavor to create any structural link between them. For example, Cloudmagic¹ creates a unified search box to access information across tools. Attachments.me² enables access to Dropbox files for creating attachments in Gmail. Neither application allows users to ‘group’ multiple open windows or tabs, the user is still left to manage those.

2.2 Task Management

Another area of research related to our work is Task Management, as knowledge workers typically have a list of pending actions for each project. Bellotti et al. (Bellotti et al., 2004) studied task management to inform the design of a task list manager. In their work, they suggested that a task manager should support *informal priority lists*, to ensure near-term execution of priority actions. Furthermore, tasks within each project can help knowledge workers prioritize and maintain their attention over different projects (Bellotti et al., 2004; González and Mark, 2004). Tasks can also act as good reminders when they appear *in the way* and always visible in the working space (Bellotti and Smith, 2000; Bellotti et al., 2004; González and Mark, 2004).

2.3 Quick Capture

Knowledge workers often capture information and tasks while working and using different tools. Bergman (Bergman et al., 2003) suggested that capturing the context of an information item during interaction assists the user to recall the information when it is later engaged with. Jones et al. (Jones et al., 2008) implemented quick capture as part of the personal project planner, where users could create rich text *project plans* and reference documents, email

¹<https://cloudmagic.com>

²<https://attachments.me>

messages, web pages, etc. Hanrahan et al. (Hanrahan et al., 2011) added a quick capture ability within the email client to move information to wikis where users could drag and drop content into a shared wiki space. Quick capture also exists as a feature in many commercial Getting Things Done (Allen, 2002) tools, such as OmniFocus³, Things⁴, and RememberTheMilk⁵.

2.4 Multitasking and Interruptions

One of the typical characteristics of today's knowledge workers is that they are routinely interrupted, as a result workers are constantly multitasking and switching projects.

Czerwinski et al. (Czerwinski et al., 2004) performed a diary study with knowledge workers to characterize how they interleave multiple tasks amidst interruptions. They found that knowledge workers switch tasks a significant number of times, with an average of 50 shifts over the week. The projects that were returned to were more complex, significantly lengthier in duration, and were rated more difficult than shorter-term projects.

González and Mark (González and Mark, 2004; Mark et al., 2005) also found that knowledge work is highly fragmented, where workers spend an average of three minutes on a task and an average of 12 minutes on a project. They found several ways in which workers manage their information to handle constant switching, including aggregating a project's different types of information into a single artifact.

2.5 Activity Based Computing

Research in activity-based computing explores how to find a better mapping of real life projects to computing systems. González and Mark (González and Mark, 2004; Mark et al., 2005) introduced the concept of *working spheres* to explain how knowledge workers conceptualize and organize their basic units of work. Volda et al. (Volda and Mynatt, 2009) created an activity-based system where they linked organization of application windows and documents of a project by associating files saved on the desktop of a virtual space with the currently active project.

2.6 Window Management

Rooms (Henderson and Card, 1986) introduced the concept of virtual spaces, which is now a part of win-

³<https://www.omnigroup.com/omnifocus>

⁴<http://culturedcode.com/things/>

⁵<http://www.rememberthemilk.com>

dow management systems of modern operating systems. Better management of space and sessions has also been explored for the web browser. Rajamanickam et al. (Rajamanickam et al., 2010) created a task-focused web browser, where web pages were grouped into tasks. Morris et al. (Morris et al., 2008) created SearchBar, a tool that stored users' search query and browsing histories, to support task resumption across multiple sessions.

Multitasking Bar (Wang and Chang, 2010) incorporated the task concept into the browser providing a browser bar with a tab for each project. Jhaveri and Rähä (Jhaveri and Rähä, 2005) created a prototype tool called Session Highlights to aid cross-session task continuation. Mozilla Firefox now has the concept of *Tab Groups*, to group together similar tabs under a single label.

For web-based information systems, management of tabs and windows is a necessity. Suspending and resuming a task is problematic as it often requires either saving or reopening several independent pages from different websites.

3 DESIGN

Based on the state of the art for the various research domains, we designed our tool with the following principles in mind:

- knowledge workers organize information into projects (González and Mark, 2004; Mark et al., 2005);
- tools like email crosscut projects (Jones and Anderson, 2011);
- information is fragmented across different applications (Bergman et al., 2006; Boardman and Sasse, 2004);
- there are structures replicated in collections (Boardman et al., 2003); and
- users need an easy way to capture and restore the state of projects (Czerwinski et al., 2004; González and Mark, 2004; Mark et al., 2005)

A full description of the tool and the implementation details is available online at (Ahuja, 2013)⁶.

3.1 Projects

A project in *Contextinator* is a collection of the browser tabs opened in the same window, as well

⁶MS Thesis available at <http://vtechworks.lib.vt.edu/handle/10919/23120>

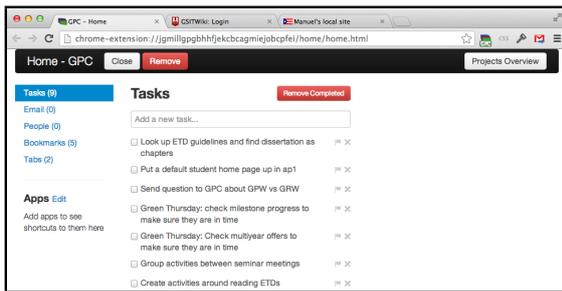


Figure 1: Project Home Page.

as a series of tasks (todo items), bookmarks, people (emails), and a series of links to external applications. Each project has a *Project Homepage* (see Figure 1) where project artifacts can be managed, for example the user can manage tasks, tabs, bookmarks, and links to external applications.

When a project is first started it begins as an empty browser window by creating the ‘File > New Window’ command from Chrome. Any tabs that are added to this window, either by adding through the ‘File > New Tab’ or just opened via user control (e.g. with a pop up menu using the Open Link in New Tab) are automatically captured as part of a project. A project state is saved automatically and does not require the user to provide a name.

Switching between projects is done by activating a different browser window. *Contextinator* saves the state of all windows, including all of the tabs opened and allows the user to switch between them. If a user closes a window, *Contextinator* can open it again restoring all of the tabs that were part of the project.

Users are able to see a preview of all their currently open projects and switch between them using either the Quick Switcher page (see Figure 2). Quick Switcher is similar to the approach taken in Gionarta (Voida et al., 2008). As projects are just regular Chrome windows, users can also use any action from the operating system’s default window manager, such as minimizing a window or using Mission Control on OS X.

Task resumption is enabled through the combination of these features, in that we preserve the state of a project whenever it is closed and reinstate the previous state whenever the project is opened again.

Global Overview

Each project also has pending tasks and related emails. *Contextinator* provides access to these in each project’s home view. We also provide a global overview of all the user’s todo and email (see Figure 3) organized by projects. In the global overview window, users are able to see any unread email,

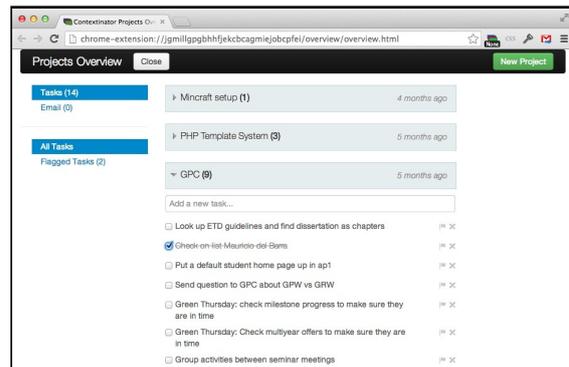
Figure 2: The *Quick Switcher* showing two projects.

Figure 3: Global overview showing the tasks across all projects. This screenshot shows three projects with two of them collapsed.

work with any tasks, and directly switch to a specific project homepage.

3.2 Information Views

As users often have accounts in multiple systems where they store different collections of data and new or different services are frequently emerging, we found the need to have an easy way to incorporate new services into our tool. For example, a label in Gmail might be related to a notebook in Evernote and a folder in Dropbox. To support the grouping of data in these independent collections, we allow each project to have an information view for each of these external services. We accomplish this with *Information views*, which provide a way to organize related information across different tools under a single project to reduce information fragmentation.

An information view is a unique URL that points to an internal location in an online collection. For example, a direct link to a folder in Dropbox produces, after user authentication, a view to that folder in Dropbox. The direct link eliminates the need for navigating to the folder within the Dropbox collection.

Contextinator stores links to external services and calls them ‘information views.’ Currently the software supports five external services, but there is no specialized code for these. In general, any online service that has a unique URL to an internal location can

be easily incorporated into a *Contextinator* project.

In addition, if users visit the application (e.g., Dropbox) directly in the web browser while having a *Contextinator* project open, they are automatically redirected to the project's *information view* eliminating redundant navigation across tools (Bergman et al., 2006; Boardman et al., 2003).

A special case of this approach is the information view for email, as email plays a significant role in project management (Bellotti and Smith, 2000; Bellotti et al., 2004; González and Mark, 2004). In order to provide a way to integrate email with projects in *Contextinator*, we display a filtered view of the email inbox. The filter shows only unread messages from the people that are part of the project and allows the user to quickly follow an ongoing email conversation without switching context to another program (email in this case). In addition, we also provide a direct link to particular Gmail tag or folder.

4 METHOD

Evaluating personal information tools and practices presents several challenges. PIM is by definition personal (Kelly and Teevan, 2007). The strategies that users follow tend to be very personal and specific to the attributes of their own collections. Thus it is very difficult to create a series of reference tests that can be *natural* to all users. Controlled lab setting do not accurately reflect the reality of the use of the technology in PIM settings. Several alternatives have been proposed, from using diary studies (Czerwinski et al., 2004; Teevan et al., 2004), *in-vivo* research methods where researchers observe users with their own information, as well as deploying a tool and collect data from its use, as done by Whittaker et al. (Whittaker et al., 2011).

To study *Contextinator*, we employed a method similar to Whittaker et al. (Whittaker et al., 2011) where they investigated email usage by deploying a program within an organization and collecting data on its use. In our work, we deployed *Contextinator* in two stages: first, we deployed the tool to a set of test users where we logged usage and later interviewed them; and second, we deployed *Contextinator* (without logging) in the Chrome Web Store where anybody could download it and install it. Several months later, we surveyed all users. This study was approved by the University IRB (#13-008). This combination of evaluation methods gives us a access to data reflecting a variety of users experiences and behaviors. We describe our research methods in stages.

4.1 First Stage: Limited Deployment

For our first stage, we recruited participants through local listservs used by computer science graduate students and faculty. We also announced the experiment in an undergraduate class where they were offered extra credit for participation. The invitation contained a URL⁷ where participants could download and install the tool. The website also included several videos explaining the use of the tool. Upon installing the tool, participants were required to agree to an online consent form.

During this stage of the evaluation we logged information about user interactions and recorded the majority of user actions with the tool (e.g. creating a new project, switching to a project, closing a project, creating a new task, flagging a task, marking a task as completed, creating a new bookmark, opening email, etc.). Each log item included the time stamp and relevant information about the event (e.g. the project or task name). We also conducted a semi-structured interview where we asked broad questions and followed up with specific questions about different areas of the tool (projects, tasks, information fragmentation, and tool usability).

Stage one provided us with detailed and rich data in regards to the use of the tool. At the same time, we interviewed a few heavy users to learn how they made use of the tool. Of the 30 participants that installed our tool, roughly one third of these were undergraduate students and the rest were graduate students. Of these 30 participants 15 of them used the tool a significant amount, the remaining 15 only created one or two projects named 'testing' or 'something.' As such, we decided to not use the data of the later group in our analyses. In the group of 15 active participants we identified 7 heavy users that created three or more projects. We interviewed 4 of our 30 participants, out of which 3 (U1, U2, U3) said they considered themselves to be heavy users of the tool and 1 (U4) that said they did not use the tool very much.

4.2 Second Stage: Broader Deployment

After our first evaluation we released *Contextinator* in the Chrome Web Store for free. At the time of this writing, over 3000 users have installed *Contextinator* and there are 20 comments in the Chrome Web Store for *Contextinator*. The project is also available on GitHub where it has 48 'stargazers', 9 'watchers', and has been forked 7 times.

In addition, we presented a survey to users of the tool upon their next activation of the software. The

⁷<http://contextinator.cs.vt.edu>

survey was to gather feedback and to clarify ideas gathered on Stage 1. In particular, we wanted to get a better sense of what users of the tool thought of the notion of ‘projects’. It is worth noting that these users are not affiliated with our institution nor connected to our research group. Only those users older than 18 years old were surveyed, a restriction of our local IRB.

Stage two provided us with a much broader population and allowed us to gather information from real users of the tool. As of this writing, of the 20 comments on the Chrome Web Store, 14 are positive, 5 are negative, and 1 is neutral. Five of the messages are bug reports and 9 are requests for new features. Each of the messages with requests had more than one requested feature.

Overall, our evaluation provided a rich and varied collection of data. We have more than 3000 users that explored our tool, very rich user logs of about 7 users, interviews with 4 users, have more than 30 survey responses, and about 60 people that have either commented or followed our project in the two online repositories (Chrome Web Store and Github). The next section presents the results and our analysis of the evaluation.

5 RESULTS

5.1 Project Appropriation

In our analysis of how our participants and users made use of the project metaphor we found two interesting aspects. First, the scope of projects varied widely, both between and within users’ projects. Second, users thought quite differently about what a project was, again, both between and within users’ projects. These different scopes and appropriations of the project metaphor point to our naive assumption that ‘projects’ is the proper and complete metaphor for managing work.

First, we see the different scope, or granularity, of projects in the number created by the participants of the first stage ($\bar{x} = 4.5$, Figure 4). When we asked users in our second phase how many projects they currently had in *Contextinator*, they reported an average of 8.59 projects ($\sigma = 7.13$). Of the 18 responses in our second stage, the maximum number of projects was 28, the variability in the number of projects among the users is illustrated by the high standard deviation in relation to the mean.

The different scopes that were used is more clearly illustrated in the names that participants in our first phase chose for their projects. Some projects are

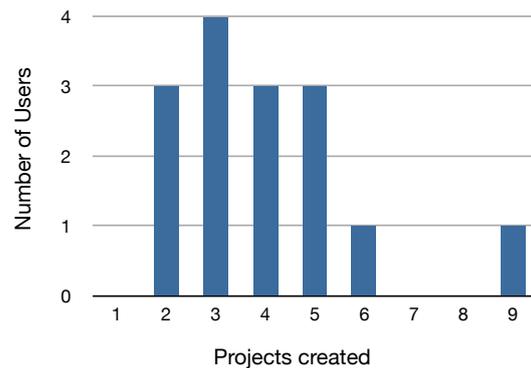


Figure 4: Distribution of number of projects created by users in *Contextinator* (stage 1).

clearly bounded as a specific task addressed, e.g. “Crypto project” and “German HW.” While there are other projects that, while also bounded, provide a grouping and represent an activity that will contain multiple *projects* (at least as we envisioned them), examples include “Algorithms” or “CS 3744”. There are even more broad projects that center around interests such as “Gardening,” as well as, projects that are even more general, e.g. “Life” or “General.” Examples of project names are shown below.

- *Individual Projects*: “Crypto project” “3114 Project 3”, “Tax Returns” and “German HW”
- *Groups of Projects*: “Algorithms”, “Usability”, “CS 3744”.
- *Ongoing Activities*: “Gardening”, “Web Development”, and “Shopping”.
- *Catch All*: “Life” and “General”

Probing further into how participants and users thought about a project gives further insight into the inadequacies of the project metaphor. During our interviews the confusion over the definition of a project was clear. Not surprisingly, the most confused participant identified themselves as not using the tool much.

When I was first using it, the title project. . . it made me feel like it should be like a school project or a research project [. . .] It definitely threw me initially (Participant U4).

A user from the Chrome Web Store, explained that they thought of projects as *contexts* instead of projects:

I’d rather call it context. A context can be a project I am working on, a research topic or an otherwise combined series of tabs. For example: Everything related to geocaching, traveling. Some contexts only exist for the duration of the ‘project’, some I keep indefinitely.

A second user from the Chrome Web Store conveyed a similar sense on their view of a project.

A set of related tabs that I need open at the same time – this means I might have multiple Contextinator projects for different stages of the same Project.

Both of these users consider projects not as a related set of work items, but more as a contextual capture of their current goals. Those goals might be immediate or longer term. The ‘projects’ in these two examples, are a more expansive view of organizing information. This sentiment was echoed by a participant in our test group where they also replaced the *project* metaphor with their own more flexible organization.

I have pretty broad categories. I have a General that I just throw stuff in. I have Web Development, so any time I am looking up stuff on stack overflow. I have Shopping, for different stuff I am shopping for. . . (Participant U2).

Another participant thought of projects more as *lists of things* to do. Yet another participant simply created projects for all his classes, enabling him to enter a context in his browser as he entered the corresponding physical context of the classroom:

The first thing I did was make a project for each class I am in. So, when i am in class, I can just open that project and have all the tabs. Esp. for Dr. XX’s class, there is like Moodle, Piazza, etc. (Participant U1).

However, regardless of what projects were to the user, nearly all of our participants found the ability to capture and resume the state of a window useful. This is illustrated by an additional appropriation of projects as a “bookmark for an entire window.”

I think of them as a bookmark for an entire window. In Chrome if you have a bunch of tabs open, and if I want to come back to all of them at once, bookmark it, make it a project (Participant U1).

Another participant found that projects enabled improved tab organization, and they began grouping their browser activity in a way that they had not previously.

I would have 50 tabs open in my one window. This really helps to have 10 or less in five windows. It is really nice (Participant U2).

5.2 Project Transitions

In designing our system we also imagined that our users would purposefully initiate clean transitions between projects (a context switch). However, in our

analysis we found that there was a somewhat clear separation between users that *purposefully* switched projects and users that *found* themselves in a project.

In our interviews (stage one), three of the participants said that they did not use the tool to decide which project to work on, but it still made it easier to work on multiple projects at once. These participants switched projects in an emergent way, that is they found they needed to switch to or create a different project once they already had a few tabs open.

A lot of the times I would just open a bunch of new tabs, and not necessarily look for an existing project first[. . .] So right now I am not in a project. And then I start googling something, and I have five tabs open. And then I realize, actually this should really go into the VTS project. . . (Participant U1).

We asked users from the Chrome Web store: “Why (or when) do you create a project?” The 18 responses can be grouped into two categories. The first group (10 responses) created projects before they started working on said ‘project.’ This group had a notion of project that was related to a goal, as if planning for work to be done ahead. The second group (8 responses), however, used projects as a way to capture work done so far but not yet completed. This group used projects as a way to suspend work to be resumed later. The goal of the project was not particularly important in the creation of the project itself.

5.3 Project Planning

In the first stage of our evaluation, 12 participants created at least one task. Overall, they created an average of 5.9 tasks ($\sigma = 9.3$). They completed 3.2 tasks on an average ($\sigma = 6.7$). Participants rarely used flagging. On an average, each participant flagged 0.8 tasks ($\sigma = 1.9$). Participants quickly captured (added a URL or note to a task) 2.8 tasks on an average ($\sigma = 4.5$).

We saw roughly two approaches to project planning that mirrored the motivations cited by users for creating projects. The first, is a more *preparatory* approach (Whittaker et al., 2011), where the user creates tasks ahead of time and completes them over a longer period of time. An example of this is a user creating a new project, quickly followed by the creation of several new tasks, and in a later session marks them as completed. This approach mirrored the users that created projects before they began working on that project.

The second approach was more *opportunistic*, where a user creates new tasks as and when required, marking them as completed in the near future (usually

in the same session). Here, the user does not have a specific planning phase of their session and instead plans and captures in situ. This approach mirrors the group of users that used projects to capture work that has not been completed yet.

5.4 Usage

We also asked users “Which of the following features are indispensable for your use of Contextinator?” Only one participant selected the ‘Task manager’, 5 selected ‘Save and reopen projects’ and 13 selected ‘Browser tab management.’ Clearly the support of managing the windows/tabs in the browser is the most used feature in our tool.

The last four questions were a likert-scale questions about their agreement/disagreement with factual statements. The results are presented below. The choices were (with score values in parenthesis): Strongly Agree (1), Agree (2), Neither Agree nor Disagree (3), Disagree (4), and Strongly Disagree (5).

“With Contextinator, I am able to work on multiple projects simultaneously.” Average 2.00 (Agree) and standard deviation of 0.7.

“With Contextinator, switching projects makes me lost, so I avoid switching unless it is absolutely necessary.” Average of 3.5 (close to neutral) and standard deviation of 0.9.

“With Contextinator, suspending or closing a project is easy because I don’t have to worry about losing data.” Average of 1.89 (Agree) and standard deviation of 1.0.

“With Contextinator, resuming a project is easy as I am able to quickly gather where I left off.” Average of 1.89 (Agree) and standard deviation of 0.6.

Based on these results, it is clear that the support for grouping related items, suspending and resuming work are the most salient features of *Contextinator*.

6 DISCUSSION

In this paper, we have presented the design and evaluation of *Contextinator*, a system built to assist people in managing their personal information stored in the cloud. The design of *Contextinator* focused on providing users support in three areas. First, it allowed users to have their web related project information in one place (including emails, bookmarks, todos, people in a project, etc.). Second, it provided a way to group and manage windows and tabs as a single project. Finally, the tool provided a way to capture, save, and reopen a project. While there is support

for all of these features, our several rounds of evaluation only found strong evidence in favor of the third, the management of opening/closing projects. This we consider the “killer feature” of *Contextinator*. This was verified in our multi-method evaluation as it was routinely mentioned in our several rounds of evaluation and confirmed by the online comments and the survey to the current users.

The management of multiple tabs in a ‘project’ proved to be very valuable. Most users found this idea so compelling that they ranked the management of context switching within our tool the most valuable feature. Being able to organize their activity with tabs and being able to stop and resume work seems to successfully address the fragmentation that naturally occurs on the web as users access multiple websites for information.

We realize that we failed to address information fragmentation as it relates to users’ social circle. Several users wanted to be able to access the context information from another computer⁸. Participant U4 (in stage one) used two computers (desktop and a laptop) interchangeably, and wanted to be able to link them and have the same projects at both places.

In addition, two participants wanted to be able to share their projects with groups of people, and be able to accomplish tasks in a project together with their collaborators. Clearly the information fragmentation is not just across information silos and devices, but also across collaborators.

Thus, we can say that we solved the information fragmentation that we set about to study only partially. The project management features were clearly well received and might account for the broad use of the tool in the Chrome Web Store. Users liked being able to group tabs as a single unit and being able to save that group and reopened it later. The other set of features (e.g., integration of information with email, task and bookmarks) not only did not seem to get much comments, we found little evidence that users used it.

6.1 Implications for Future Work

With our study we also gained insights into other areas that might allow future researchers and developers to better address the problems we explored. We would like focus on two of them here. First is the idea of what a user considers a ‘project.’ The second is how ‘information views’ address information fragmentation.

⁸This feature is now part of the tool

6.1.1 What is a project?

First, we provide some insight onto what users consider a ‘project.’ People have vastly different concepts as to what a ‘project’ is and designs of similar tools should provide for this variability. A system that seeks to improve support for knowledge work should be able to blend in with this existing ecosystem of tools.

There is a wide variability of what users call a project. Typically we consider a project as a set of related activities with a particular goal in mind. But we found that users have a much broader definition. In some cases, users considered several ‘projects’ as all part of the same task at hand, thus requiring multiple active projects at once. In those cases, one or more of the projects were really collection of resources that were reused in similar tasks (e.g., having a project with reference websites for web development).

In addition, we have identified two types of ‘projects’ based on why and when users create them. ‘Preparatory’ projects are projects that allow users to organize their work, including creating tasks to be performed later in time. These projects are often created a priori of the work to be done. The second type of project, ‘opportunistic,’ are projects that emerge from work that is being done. These projects might or might not have a specific goal, and instead emerge from users’ work. Opportunistic projects seem to benefit from capturing the context of work for suspension and resumption of work.

6.1.2 Information Views

The second lesson learned is how ‘information views’ addresses the problem of information fragmentation. In our tool, we had three ways to integrate information.

The first and most simple form for information views to address information fragmentation is to develop a program that actually supports multiple sources of information all into one service. We did this for the collection of todos, bookmarks, and people. All of the information for those three categories were collected by our tool and stored locally in our tool. The problem with this approach is one of adoption. In our data, we found that most users did not use this feature. In the use logs that we captured, we found that most users did not even configure this feature.

The second approach, exemplified by our presentation of email in the project page, is to show enough information from an external source without requiring going to other websites. This is the example used in the web today of ‘content embedding.’ Google Maps,

for example, allows users and developers to embed a map view in other sites. Instead of having a link and requiring navigation to another site/tool, the information is presented in place with some minor restrictions. Our tool did this by presenting unread emails in the context of a project. This approach requires external services to provide access to their data via some API or protocol (e.g., IMAP access to email).

The last one and the one that holds the most promise is to make use of direct links to outside repositories thus providing in different tabs direct access to related information. Our tool captured a direct link in each project to an external service (e.g., a link to a folder in Dropbox). These links are then presented as short-cuts in the project home page. But more interesting, within the context of the project, clicking on a generic top level to an external service is automatically redirected to the internal location for the project. The only requirement to integrate external tools with *Contextinator* is to provide unique URLs to its internal data.

These three approaches to building integrated information views allow us to at least begin to address the information fragmentation that occurs on the web. A tool like *Contextinator* has the potential to create (or re-create) the context lost amid information fragmentation in today’s web-based tools.

REFERENCES

- Ahuja, A. (2013). *Contextinator: Recreating the context lost amid information fragmentation on the web*. Masters thesis, Department of Computer Science, Virginia Tech.
- Allen, D. (2002). *Getting Things Done: The Art of Stress-Free Productivity*. Penguin Books.
- Bellotti, V., Dalal, B., Good, N., Flynn, P., Bobrow, D. G., and Ducheneaut, N. (2004). What a to-do: studies of task management towards the design of a personal task list manager. In *Proc. CHI '04*, pages 735–742, New York, NY, USA. ACM.
- Bellotti, V. and Smith, I. (2000). Informing the design of an information management system with iterative fieldwork. In *Proc. DIS '00*, pages 227–237, New York, NY, USA. ACM.
- Bergman, O., Bergman, O., Beyth-marom, R., and Nachmias, R. (2003). The user-subjective approach to personal information management systems. *Journal of the American Society for Information Science and Technology*, 54:872–878.
- Bergman, O., Beyth-Marom, R., and Nachmias, R. (2006). The project fragmentation problem in personal information management. In *Proc. CHI '06*, pages 271–274, New York, NY, USA. ACM.
- Boardman, R., Boardman, R., Spence, R., and Sasse, M. A. (2003). Too many hierarchies? the daily struggle for

- control of the workspace. *Proc. CHI '96*, pages 406–412.
- Boardman, R. and Sasse, M. A. (2004). "stuff goes into the computer and doesn't come out": a cross-tool study of personal information management. In *Proc CHI '04*, pages 583–590, New York, NY, USA. ACM.
- Czerwinski, M., Horvitz, E., and Wilhite, S. (2004). A diary study of task switching and interruptions. In *Proc. CHI '04*, pages 175–182. ACM.
- González, V. M. and Mark, G. (2004). "constant, constant, multi-tasking craziness": managing multiple working spheres. In *Proc. CHI '04*, pages 113–120, New York, NY, USA. ACM.
- Hanrahan, B., Bouchard, G., Convertino, G., Weksteen, T., Kong, N., Archambeau, C., and Chi, E. H. (2011). Mail2wiki: low-cost sharing and early curation from email to wikis. In *Proc. C&T 2013*, pages 98–107, New York, NY, USA. ACM.
- Henderson, Jr., D. A. and Card, S. (1986). Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Trans. Graph.*, 5(3):211–243.
- Jhaveri, N. and Riih a, K.-J. (2005). The advantages of a cross-session web workspace. In *Proc. CHI EA '05*, pages 1949–1952, New York, NY, USA. ACM.
- Jones, W. and Anderson, K. M. (2011). Many views, many modes, many tools ... one structure: Towards a non-disruptive integration of personal information. In *Proc. Hypertext 2011*, pages 113–122, New York, NY, USA. ACM.
- Jones, W., Klasnja, P., Civan, A., and Adcock, M. L. (2008). The personal project planner: planning to organize personal information. In *Proc. CHI 2008*, pages 681–684, New York, NY, USA. ACM.
- Jones, W., Phuwanartnurak, A. J., Gill, R., and Bruce, H. (2005). Don't take my folders away!: organizing personal information to get things done. In *Proc. CHI EA '05*, pages 1505–1508, New York, NY, USA. ACM.
- Karger, D. R. and Jones, W. (2006). Data unification in personal information management. *Commun. ACM*, 49(1):77–82.
- Kelly, D. and Teevan, J. (2007). *Personal Information Management*, chapter Understanding What Works: Evaluating PIM Tools, pages 190–204. University of Washington Press.
- Mark, G., Gonzalez, V. M., and Harris, J. (2005). No task left behind?: examining the nature of fragmented work. In *Proc. CHI '05*, pages 321–330, New York, NY, USA. ACM.
- Morris, D., Ringel Morris, M., and Venolia, G. (2008). Searchbar: a search-centric web history for task resumption and information re-finding. In *Proc. CHI '08*, pages 1207–1216, New York, NY, USA. ACM.
- Rajamanickam, M. R., MacKenzie, R., Lam, B., and Su, T. (2010). A task-focused approach to support sharing and interruption recovery in web browsers. In *Proc. CHI EA '10*, pages 4345–4350, New York, NY, USA. ACM.
- Teevan, J., Alvarado, C., Ackerman, M. S., and Karger, D. R. (2004). The perfect search engine is not enough: A study of orienteering behavior in directed search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '04*, pages 415–422, New York, NY, USA. ACM.
- Voida, S. and Mynatt, E. D. (2009). It feels better than filing: everyday work experiences in an activity-based computing system. In *Proc. CHI '09*, pages 259–268, New York, NY, USA. ACM.
- Voida, S., Mynatt, E. D., and Edwards, W. K. (2008). Reframing the desktop interface around the activities of knowledge work. In *Proc. UIST '08*, pages 211–220, New York, NY, USA. ACM.
- Wang, Q. and Chang, H. (2010). Multitasking bar: prototype and evaluation of introducing the task concept into a browser. In *Proc. CHI '10*, pages 103–112, New York, NY, USA. ACM.
- Whittaker, S., Matthews, T., Cerruti, J., Badenes, H., and Tang, J. (2011). Am i wasting my time organizing email?: a study of email refinding. In *Proc. CHI '11*, pages 3449–3458, New York, NY, USA. ACM.

SHORT PAPERS

SafeMash

A Platform for Safety Mashup Composition

Carlo Marcelo Revoredo da Silva, Ricardo Batista Rodrigues,
Rafael Roque de Souza and Vinicius Cardoso Garcia
*Informatics Center, CIN, Federal University of Pernambuco, UFPE, Av. Jornalista Aníbal Fernandes, s/n,
Cidade Universitária, Recife, Brazil
{cmrs, rbr, rss4, vcg}@cin.ufpe.br*

Keywords: Web Mashups, Security Policies, Information Flow.

Abstract: This article describes the SafeMash project, a platform that provides an environment for the construction, safe consumption and standardized of Mashups. The platform proposal is to offer functionalities focused in security aspects regarding the integration between web applications, the users and third parties APIs. Which is based in one specification to build an standardized Mashup. Those resources are based in security approaches specified by organizations such as OWASP and OpenMashup Alliance.

1 INTRODUCTION

With the advent of the Web 2.0, the content produced and published was no longer for only reading: now the users interact sharing information on sites and services that make the data available, making the web programmable, allowing that computers and humans work on a cooperative way (Governor et al., 2009).

In this context, arises the Mashups, applications able to combine existing functionalities, data and interfaces in order to reach a more complex goal. The data from heterogeneous sources are integrated, normally available through Web Services. The result of the integration of all service invocations may, for example, be presented in a site with usability resources, through technologies such as the Extensible Markup Language (XML) and Asynchronous Javascript and XML (Ajax) (Bozzon et al., 2009).

On the other hand, one of the obstacles is the lack of standardization, besides; problems related with security must also be highlighted. According to (Allen, 2001), we may relate Information Security with the protection of the set of information aimed to one organization or user, based in characteristics of Confidentiality, Integrity and Availability (CIA), being applied to all aspects of information and data protection.

Also according to (Allen, 2001), a Security

Policy represents a set of controls established and that must not be violated, in order to minimize vulnerabilities in APIs, preventing them from being exploited by attacks such as the Cross Site Script (XSS), which is defined as the possibility of malicious users to inject scripts in the client side, with huge consequences, such as for example changes on the graphic interface of the application, compromising the integrity of the content displayed (OWASP: XSS, 2013).

According to a group of researchers from the University of California at Berkeley, the lack of standardization of APIs is among the ten biggest obstacles to the adoption of Cloud Computing solutions (CN) (Armbrust et al., 2009).

Each Content Provider implements the API in its own way, causing problems of standardization and divergence between the security offered and desired. With this, the interfaces present with heterogeneity, which hinders the insertion of a Security Policy efficiently, making your users feel afraid to enter their information, to the extent that the data traffic are becoming increasingly valuable.

This article has the goal to describe a platform called SafeMash, presented as an architectural model that may be applied in any domain that wishes to execute Mashup compositions. The proposal is to make available an environment for three main objectives: (i) sanitize input and output data between parties involved at Mashup consumption, (ii) where users can build Mashup with standard specification,

and (iii) analyse the compliance of security policies intended by Mashup services.

This work is presented with the following structure: in the section 2, security questions in a standard Mashup environment are described; in the section 3 the proposal of our methodology is presented, and partial result of an objective of our proposal. Finally, in the section 4 final conclusions and future goals are described.

2 MASHUP ECOSYSTEM AND SECURITY ISSUES

The work is focused in the communication flow between Mashups and other parties involved. For a better understanding of the problem, this information flow will be described, in this section, being represented as an ecosystem. And finally, the goal is to research some relevant aspects about information security in those scenarios.

2.1 The Mashup Ecosystem

The goal of Mashups is to collect several data through one or more APIs, which are inherent to third party services, known as Content Providers (CP). Those data will be transformed in a combined content, which will be presented as an answer to the Mashup Consumer (MC) requisition. As an example scenario, the MC may be a web application that requires content through a specification made available by the Mashup site and waits, as an answer, a graphic, functional component, adaptable to its graphic interface.

Therefore, being able to interact with user and application, this kind of component is called a Widget (Wilson et al., 2012). In Figure 1, a workflow of a Mashup ecosystem is illustrated, where the application makes a requisition and receives, as the answer, a Widget.

2.1.1 The Consumer

For a better understanding, consider that the web application in the example is a webmail, that has in its first page a widget that presents itself to the user as a calendar, coming from an external service called MashCalendar, which is used by the user to manage its appointments.

The webmail makes available a section where the user configures the widget, informing its access credentials to the MashCalendar service. After confirming the sending of information, the user is

redirected for the main page. It is in this moment that the webmail starts its role as a Mashup consumer.

2.1.2 The Intermediate

Following the flow, the webmail will communicate through a specification based interface (Mashup specification), previously agreed by the managers of the MashCalendar service and will demand the calendar for the Mashup application of the MashCalendar, which will make requisitions with one or more CP distributed in the web.

For each CP, it will be an obligation of the Mashup application, that is, of MashCalendar managers, the implementation of the communication with each CP respective API.

2.1.3 The Third-Party

Each CP will answer its requisition with a given data. For example, the application Mashup will demand information of national or regional holidays, maps describing the location and transit on the user displacement to each scheduled meeting, among other information. Because of that, the Mashup site will combine those answers, generating this way the content that the webmail application waits as an answer, which will result the calendar with the appointments scheduled by the user.

2.2 Security Issues

Based in this example is possible to consider some worrying aspects in relation to Security. The first is when the user gives to the webmail his MashCalendar credentials, which will need to make available for the webmail a specification that allows reliability and integrity of the information that will flow.

An important aspect is related to the answer that will be generated by MashCalendar, the webmail must be sure that it will not bring information and behaviors that might harm the user. Such as XSS or other attacks such as: Cross Site Request Forgery (CSRF), which consists in inserting malicious requisitions in a browser session opened by the user, allowing it to be stolen, and in that way compromising its credential confidentiality (OWASP: CSRF, 2013). Another potential attack vector for environments with graphical user interface is an attack known as clickjacking, is when an attacker uses multiple transparent or opaque layers that overlap one form to trick the user had the intention of clicking a link or button that is overlaid

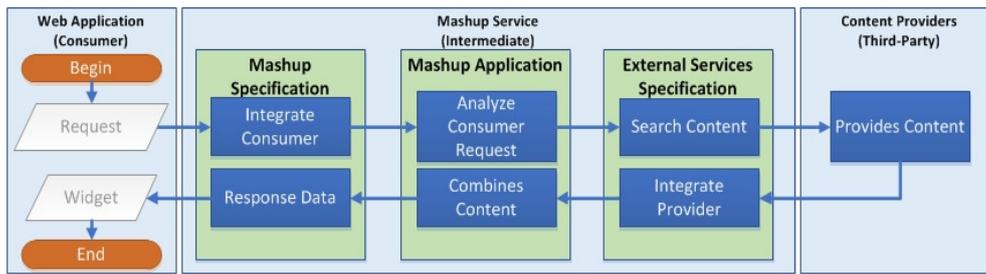


Figure 1: Mashup Ecosystem Workflow.

With this, the attacker "hijacks" user click, redirecting it to another application or domain. Nothing prevents this technique is also used to hijack clicks of a digital keyboard, often used in internet banking (OWASP: Clickjacking, 2013).

These attack vectors are exploited vulnerabilities in browsers. Every browser has a security layer in its architecture, which is based on behavior policies. One of these is known as the same-origin policy, that it is a practice where it inhibits JavaScript (JS) code load HTML documents, forms or frames from other domains.

However, this does not apply to the loading of other JS code, potentially able to break this policy through mechanisms that use XMLHttpRequest (Barth et al., 2008), such as jQuery (jQuery, 2013), JSONP (JSONP, 2013) and YQL (YQL, 2013).

3 THE SafeMash PLATFORM

In order to reach the goals proposed in this work, one research was made in literature about the state of the art of techniques of vulnerability detection and web applications/services. Publications from Open Web Application Security Project (OWASP) (OWASP: Top Ten Project, 2013) and Open Mashup Alliance (OMA) (Open Mashup Alliance, 2013) are considered as the fundamentals for this work and, in order to ease references, at this work will be named as specifying organizations (SO).

3.1 Users Scenarios

In Figure 2, based on the previous example, we present a workflow of same widget consumption, but running at SafeMash platform. Regarding the utilization of SafeMash, 3 user scenarios must be considered: Consumer, Developer and Administrator, they interact with 4 main components such as "Sanitization Filter", "Integration Manager", "Security Policy" and "Services Repository", which

relate between 14 main actions: "Request Content", "Build Mashup", "Search API/Mashup", "Define Policies", "Manage Mashup", "Sanitize Request", "Sanitize Response", "Combines Content", "Response Data", "Integrate Services", "Get Service Policies", "Search Service", "Process Response" and "Get Content" as illustrated in Figure 3.

3.1.1 The Consumer User

This scenario happens when one user, guest or registered, accesses the platform. In this case the user action will only have the intention to use a service or composition cataloged or built in the platform through the resource through the action "Request Content". When the content is asked for, the action called "Search Service" will be used, which will search services cataloged or built in the platform.

The user request will be monitored by the component "Sanitization Filter", where it will be analyzed with the goal to minimize attacks based in malicious injections and vulnerabilities. This layer will use a meta-model based on techniques used by the SO.

However, it is important to mention that, independently of the outlook presented, some user requests, be them in the Consumer, Developer or Administrator scenario, established by the platform itself, will be considered inexorable to be measured by this layer.

The next step will be to redirect the user request for the Mashups or services requested, where they will be analyzed by the component "Security Policies", with the goal of verifying if the solicited CP is fulfilling the specified security policy.

This layer is also based on a meta-model based in the SO recommendations. And finally, the platform makes the action "Response Content", to retrieve the required response in providers contents spread around the web.

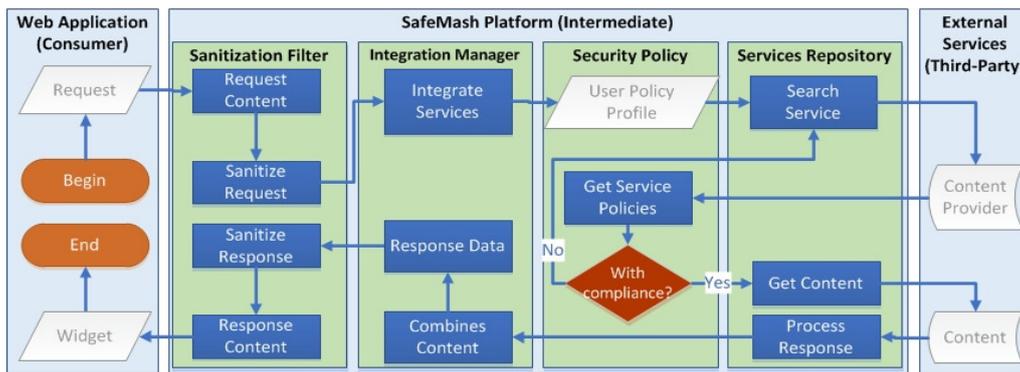


Figure 2: Mashup Ecosystem Workflow at SafeMash Platform.

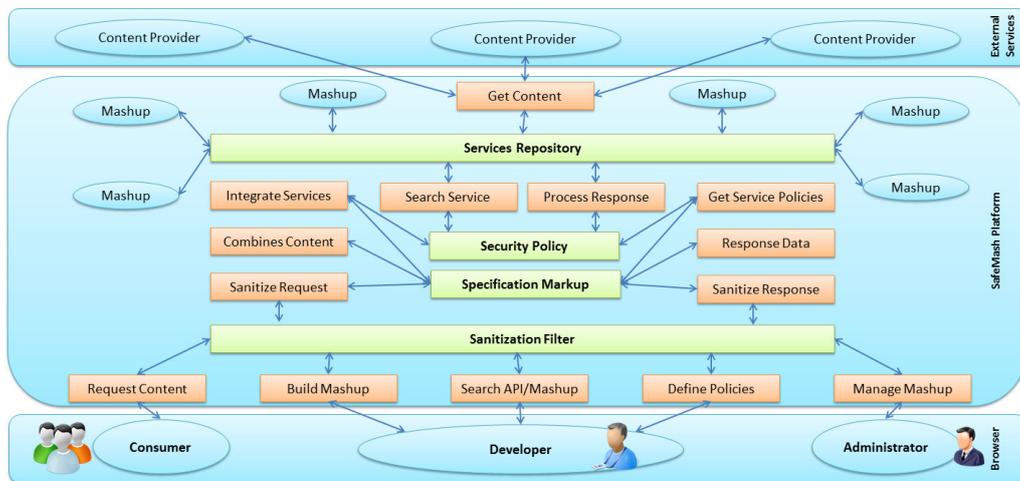


Figure 3: The main components and actions of SafeMash Architecture.

3.1.2 The Developer User

This scenario happens when the platform is accessed by a registered user that wishes to create a Mashup. Available to the user is a control panel that offers basically three actions:

The first action allows the user to build its Mashup, where it is possible to insert syntaxes through the action named “Build Mashup”, making use of the component “Integration Manager”, with syntaxes according the specification Enterprise Markup Mashup Language (EMML), created by the OMA, based in XML, with the intention to make the creation of Mashups more homogeneous.

The second action allows the user to search for available CPs through the action “Search API/Mashup”, where the user may perform a semantic filtering and search for contents, catalogued in the component “Services Repository”, that it wishes to use to compose its Mashup. These are then interpreted into action “Integrate Services”.

In this stage, it is also possible to make restrictions in which the user will specify which security resources might be considered. From this, the platform will use the component “Security Policy” in order to filter by Mashups or APIs that fulfill the specified resources. The third action allows the user to attribute a set of policies in its own Mashups, through the action “Define Policies”. In this context, it will specify all security resources applied to its composition. This information will later be analyzed and tested by the platform administrators.

3.1.3 The Administrator User

An administrator is a user with privileges to perform an analysis through the action “Manage Mashup” in each composition created by a Mashup developer. From that analysis, the proposed composition will be released or not to the public, through the component “Security Policies”.

This practice will contribute to other users, that in the future will consume, to have knowledge about

which security aspects are being considered by it, thus characterizing the environment of continuous security and standardization control.

3.2 Proposal Goals

In this section, we present some practices applied on the platform, which are directed to reach 3 desired goals by our proposal.

3.2.1 Sanitize Input/Output Flow

As a practical approach, we use a development release of our component "Sanitization Filter", in order to observe the impacts to be considered when it is inserted in real environments.

We develop a simple mashup that consumes a service through the Yahoo! Weather API (Yahoo! Weather, 2013), with parameter WOEID assuming the value 26802884 (City of Recife), where our component, through action "Sanitize Response," do a read from a byte array that represents the content as response.

Upon receiving this response, called the Mashup our component that makes a filter removing possible threats in response, for example, XSS, CSRF and Clickjacking techniques as described in Table 1, and Table 2 is an example of XSS attack using one of the techniques mentioned above.

Table 1: Some techniques applied to detect vulnerabilities.

| Threat | Behavior |
|--------------|---|
| XSS | Malicious JS codes and tags; suspect code at JSON data blocks; HTML and URI encoded; Inadequate uses of eval() or JSON.parse() functions; |
| Clickjacking | Malicious embed JS; Overlap suspect elements |
| CSRF | HTTP headers suspects; Tags IMG with suspect values in "src" attribute. |

Table 2: An example of XSS attack using HTML/HEX encoding.

| |
|---|
| <pre>&#x3C;&#x73;&#x63;&#x72;&#x69;&#x70;&#x74;&#x3E;&#x61;&#x6C;&#x65;&#x72;&#x74;&#x28;&#x27;&#x58;&#x53;&#x53;&#x27;&#x29;&#x3B;&#x3C;&#x2F;&#x73;&#x63;&#x72;&#x69;&#x70;&#x74;&#x3E;</pre> |
|---|

3.2.2 Standardize Composition

The Platform users have at their disposal a set of tools, such as drag and drop elements, a declarative

language, besides a wizard composition, which will be responsible for any abstract complexity in the development of a Mashup, through action "integrate Services ", and consumption, through action "Combines Content ".

All content of the presentation layer will be transformed into EMMML, bringing an oblique language between the control layer and presentation platform. The component "Specification Notation" will be responsible for performing validations syntax and operations EMMML conversions in the code, resulting in a more specific language for the domain logic platform.

3.2.3 Promote Security Policies

The users themselves will describe what type of security practices should engage external services, such as the developer can define, through action "Define Policies" which a particular content of your Mashup can only be consumed from external service to perform the traffic content through https, or Customer may request, through the action "Get service Policies", the content at issue must be from an external service that has a well-structured policy which is committed to ensuring privacy in data traffic.

All these settings users will be managed by the component "Security Policy", and will be periodically evaluated by the Administrators of the platform, through the action "Manage Mashup".

4 CONCLUSIONS

In this article we present an environment ecosystem Mashups, describing benefits and purposes, as well as their weaknesses regarding security issues and the lack of standardization in these environments.

As proposal, we present a platform with the objective of minimizing the obstacles addressed. Currently, the proposal is in the implementation stage of its main components and documentation of architectural decisions.

As future works, the intention is to present the main non-functional requirements that must be fulfilled by the platform, in order to define the main functionalities. For the formal documentation the intention is to have Use Case Diagrams, which according to (Fowler, 2003), describe functionalities through diagrams with Unified Modeling Language (UML) notation, and that has the objective to elaborate a documentation describing functionalities in a graphic and intuitive way.

According an example in Figure 4, which describes the actions of Consumer, Developer and Administrator users, and the Behavior of the “Sanitization Layer” and “Policy Layer” components in the SafeMash Platform.

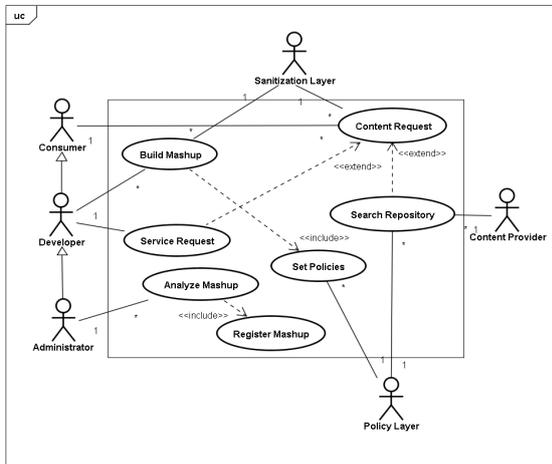


Figure 4: UML Use Case Diagram of SafeMash Actors.

Additional, we are developing two artifacts in low-level details: one which documents the main components and their relationships in the architecture, and one which specifies the layered architecture known as Layered View, which according to (Bachmann et al., 2001) provides greater flexibility in development since the architecture subdivides into distinct layers, facilitating the identification of the main features of the system and assists in the practice of reuse.

And we intend to present experimentations of the main components of the platform in real environments, using formal techniques to metric in an experiment in order to obtain satisfactory results in our goals.

REFERENCES

Allen, Julia H., 2001. The CERT Guide to System and Network Security Practices. *Addison-Wesley*.
 Armbrust et al. (2009) “Above the Clouds: A Berkeley View of Cloud”, Electrical Engineering and Computer Sciences, University of California at Berkeley.
 Bachmann et al., 2001. Software Architecture Documentation in Practice. *Addison Wesley*.
 Barth et al., 2008. Securing Frame Communication in Browsers. in: *17th USENIX Security Symposium*, p 17-30.
 Bozzon et al. 2009. A Conceptual Modeling Approach to Business Service Mashup Development, *IEEE*

International Conference on Web Services, Los Angeles, CA, USA, pp. 751-758.
 Fowler, Martin, 2003. “UML Distilled: A Brief Guide to the Standard Object Modeling Language”, *Addison-Wesley*, 3 edition.
 Governor et al. (2009) “Web 2.0 Architectures: What Entrepreneurs and Information Architects Need to Know”. *O’Reilly*; 1 ed.
 JQuery, 2013, <http://jquery.com/>
 JSONP, 2013, <http://www.json-p.org/>
 Yahoo! YQL, 2013, <http://developer.yahoo.com/yql/>
 Open Mashup Alliance EMMML Documentation. <http://www.openmashup.org/omadocs/v1.0/>, last access in April 2013.
 OWASP:Clickjacking, <https://www.owasp.org/index.php/Clickjacking>, last access in April 2013.
 OWASP: Cross Site Request Forgery (CSRF), [https://www.owasp.org/index.php/Cross-Site_Request_Forgery_\(CSRF\)](https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)), last access in April 2013.
 OWASP: Cross Site Script (XSS), [https://www.owasp.org/index.php/Crosssite_Scripting_\(XSS\)](https://www.owasp.org/index.php/Crosssite_Scripting_(XSS)), last access in April 2013.
 OWASP Top Ten Project (2010), https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project, last access in March 2013.
 Wilson et al., 2012. Orchestrated User Interface Mashups Using W3C Widgets. *Springer Berlin Heidelberg Lecture Notes in Computer Science Volume 7059*, pp 49-61.
 Yahoo! Weather, 2013. <http://developer.yahoo.com/weather/#req>.
 Moore, R., Lopes, J., 1999. Paper templates. In *TEMPLATE’06, 1st International Conference on Template Production*. SCITEPRESS.
 Smith, J., 1998. *The book*, The publishing company. London, 2nd edition.

GeoSPARQL Query Tool

A Geospatial Semantic Web Visual Query Tool

Ralph Grove¹, James Wilson², Dave Kolas³ and Nancy Wiegand⁴

¹*Department of Computer Science, James Madison University, Harrisonburg, Virginia, U.S.A.*

²*Department of Integrated Science and Technology, James Madison University, Harrisonburg, Virginia, U.S.A.*

³*Raytheon BBN Technologies, Columbia, Maryland, U.S.A.*

⁴*Space Science and Engineering Center, University of Wisconsin, Madison, Wisconsin, U.S.A.*

groverf@jmu.edu, wiegand@cs.wisc.edu, dkolas@bbn.com, wilsonjw@jmu.edu

Keywords: Semantic Web, Geographic Information Systems, GeoSPARQL, SPARQL, RDF.

Abstract: As geospatial data are becoming more widely used through mobile devices and location sensitive applications, the potential value of linked open geospatial data in particular has grown, and a foundation is being developed for the Semantic Geospatial Web. Protocols such as GeoSPARQL and stSPARQL extend SPARQL in order to take advantage of spatial relationships inherent in geospatial data. This paper presents GeoQuery, a graphical geospatial query tool that is based on Semantic Web technologies. GeoQuery presents a map-based user interface to geospatial search functions and geospatial operators. Rather than using a proprietary geospatial database, GeoQuery enables queries against any GeoSPARQL endpoint by translating queries expressed via its graphical user interface into GeoSPARQL queries, allowing geographic information scientists and other Web users to query linked data without knowing GeoSPARQL syntax.

1 INTRODUCTION

The Semantic Web has the potential to greatly increase the usability of publicly available data by allowing access to open data sets in linked format over the Web. W3C standards such as RDF (Manola 2004) and SPARQL (Prud'hommeaux, 2008) enable standard access to data stored in triple stores that are accessible over the Web at SPARQL endpoints. The Linked Open Data (Bizer, 2009) movement adds best practices for publishing data in order to maximize availability and usability.

As geospatial data are becoming more widely used through mobile devices and location sensitive applications, the potential value of linked open geospatial data in particular has grown, and a foundation is being developed for the Semantic Geospatial Web (Egenhofer, 2002). Protocols such as GeoSPARQL (Perry, 2010) and stSPARQL (Kyzirakos, 2012) extend SPARQL, the standard RDF Semantic Web query language, in order to take advantage of spatial relationships inherent in geospatial data.

In this paper we present the GeoSPARQL Query

Tool (GeoQuery)¹, a graphical geospatial query tool that is based on Semantic Web technologies. GeoQuery translates queries expressed through its graphical user interface into GeoSPARQL queries, which can then be executed against any GeoSPARQL endpoint. With GeoQuery, geographic information scientists can query linked data and see map output without knowing GeoSPARQL syntax.

1.1 Motivation

The availability of linked open geospatial data and the Semantic Web will offer the potential for enhancing the value of geospatial data to the user in several ways.

- *Data Search and Discovery*: Semantic Web protocols could be used to dynamically discover and examine geospatial datasets over the Web, so that newly created or revised datasets will be of immediate value. The geospatial features of GeoSPARQL could also allow spatial operations to be incorporated into queries over metadata contained in catalogs of available open data during

¹ <http://geoquery.cs.jmu.edu>

a search.

- *Integration of Data Sets*: SPARQL provides the ability to integrate data services within queries, so that a user can perform logical queries of data from multiple servers without regard to the details of accessing multiple datasets and their respective formats. One query, for example, might perform spatial operations on data obtained from an open map service, a public repository, and a corporate data store.
- *Potential Applications of Semantic Web Technologies*: Beyond direct geospatial queries (e.g., show me what public buildings exist within the bounds of this city), techniques associated with the Semantic Web such as ontological reasoning and machine intelligence offer the potential for smarter user interfaces that can anticipate queries and integrate geospatial reasoning into emerging technologies such as automated vehicles and location aware phones.

The last ten years have seen tremendous growth in the development and utilization of the Internet in Geographic Information Systems (GIS). This growth is taking place in standard desktop software that is able to access geospatial data, maps, and geoprocessing services through the Internet, as well as web-based frontends to these same Internet based resources.

Currently, geospatial data are just starting to be represented in RDF (e.g., Varanka, 2012, and work by the Ordnance Survey), and the full potential of geospatial data available over the Web has yet to be realized. Open standards such as those developed by the Open Geospatial Consortium (OGC)² have played an important role in facilitating these developments. Most of these developments have been an evolution of existing GIS technologies, with only minor explorations into semantic technologies. The most successful developments have been in simplifying the access to distributed environments, allowing non-specialists to take advantage of mapping on the Internet. The small forays into geospatial semantics on the Internet have mostly been useable only by experts.

Conventional GIS allow users to explore, via a graphical user interface, datasets that are stored in proprietary or specialized data storage formats. Now, with linked data in RDF, the data representation format is open and standard. The GeoSPARQL query language is a new standard with which anyone can pose queries to data over a

provided web-based endpoint. However, SPARQL and GeoSPARQL queries are not easy to write correctly without training. Prior attempts to overcome the difficulty of learning SPARQL include, for example, a visual SPARQL editor (Collustra, 2013). It might also be possible to approach this problem through a natural language interface to GeoSPARQL queries. However, natural language processing remains a difficult, not completely solved problem.

GeoQuery is a first step towards developing a universal query tool for the Geospatial Semantic Web. GeoQuery demonstrates that it is feasible to execute geospatial queries based upon the Semantic Web infrastructure using a graphical user interface. This ability is of value to geospatial professionals who need access to the data but are not trained in Semantic Web technologies. Because GeoSPARQL queries are viewable in GeoQuery, users can also learn about the GeoSPARQL language.

2 RELATED WORK

The Ordnance Survey in Great Britain pioneered Semantic Web work for geospatial data, including the use of linked data (e.g., The Linked Data Web, 2013) and the building of geospatial ontologies (e.g., Denaux et al., 2011). The Ordnance Survey held the first Terra Cognita geospatial workshop at the 2006 International Semantic Web Conference (ISWC) to add spatial data to the Semantic Web. The Spatial Ontology Community of Practice (SOCoP), along with others, have continued the series with the fifth one (Terra Cognita, 2012) being held with ISWC 2012.

In the United States, the Geological Survey (USGS) is doing work on linked data and ontologies (Geospatial Semantics and Ontology, 2013), including a recent workshop (Varanka, 2012). The USGS has translated some of The National Map data into RDF format. Transferring spatial data into RDF is a new area that the authors are also working on. Meanwhile, to handle nonspatial RDF data, leading database companies, such as Oracle and DB2, have added RDF storage and processing to their relational database systems e.g., (Das et al., 2004, Ma et al., 2008). Because query-processing of geospatial data in RDF is still new, Garbis et al. (2013) recently developed a benchmark to judge the performance of several RDF stores for geospatial querying. The database community is also interested in temporal aspects of the Semantic Web, and a bibliography has

² <http://www.opengeospatial.org/>

been compiled that also includes spatial references (Grandi, 2012).

Koubarakis et al. (2012) delineate areas of research for linked geospatial data, of which one area is user interfaces. They pose questions as to whether user interfaces should be based on natural language or be graphical, what high level APIs would ease rapid development, and whether interfaces could be built using existing platforms such as Google Maps, Bing Maps, or OpenStreetMap. In our work, we developed a graphical interface and used OpenStreetMap and Web Map Service for map display.

Our work uses the GeoSPARQL model as an extension to SPARQL. There is another spatial extension to SPARQL, stSPARQL, which is implemented in Strabon (Kyzirakos, 2012). Strabon extends Sesame, which has the ability to have PostGIS as a backend DBMS and spatial query processor. stSPARQL and GeoSPARQL do not overlap perfectly in functionality: stSPARQL includes aggregate functions and update capabilities (without which stSPARQL is a subset of GeoSPARQL), while GeoSPARQL includes an ontology and allows for topological relations as triples. A query language in addition to SPARQL and stSPARQL that incorporates spatial considerations is SPOTL (SPO + Time and Location) in the YAGO2 project (Hoffart et al., 2012). Time and location of facts are represented through reification.

3 SEMANTIC WEB TECHNOLOGIES FOR GEOSPATIAL DATA

Most early efforts to add geospatial data to the Semantic Web focused on very simple geospatial data, i.e., points represented by latitude and longitude. The W3C Geo ontology is popular for representing such points. Though this is sufficient for many domains and use cases, more complicated geospatial domains require the ability to use multiple coordinate systems and to store polygons and other shapes. This led to development of GeoSPARQL and its support in Parliament (see section 3.2).

3.1 GeoSPARQL

GeoSPARQL provides a unifying vocabulary for geospatial data on the Semantic Web. GeoSPARQL has two key parts: a small ontology for representing

geospatial entities, and a set of query functions for processing relationships between the geospatial entities. The ontology is derived from well-used and well-understood concepts from the OGC and uses much of the same terminology as other OGC standards. The ontology is intentionally small so that it can be easily understood and easily attached to an appropriate domain ontology.

There are two key classes in the GeoSPARQL ontology: Feature and Geometry. A Feature is simply any entity (physical or abstract) with some spatial location. This could be a park, airport, monument, restaurant, etc. A Geometry is any geometric shape, such as a point, polygon, or line, and is used as a representation of a feature's spatial location. A third class, SpatialObject, is a superclass of both Feature and Geometry.

A Feature has only one primary property, *hasGeometry*. This property links the Feature to a Geometry that represents where it is in space. A Feature can have multiple Geometries, in which case it may specify one of these as the defaultGeometry to be used for spatial reasoning.

A Geometry has a number of properties, but the most important ones are those that relate the Geometry to a concrete spatial representation. These are *asWKT* and *asGML*, depending on whether the representation is in Well Known Text (WKT) (Open Geospatial Consortium, 2011) or Geography Markup Language (GML)³ respectively. The properties point to an RDF literal with a data type of *wktLiteral* or *gmlLiteral*. Within these literals are the points that delineate the geometry: for example, the corners of a polygon.

The general usage of this ontology is to attach it to the ontology of the domain. If a domain ontology includes classes with relevant geospatial locations, those classes are declared subclasses of Feature. In this way they inherit the *hasGeometry* property and its link to the Geometry class.

The query functions in GeoSPARQL are used to relate the Features and Geometries to one another. The functions include binary topological relationships, set combinations of Geometries (ex. union, intersection), and other calculations such as distance. When possible, GeoSPARQL provides multiple sets of terminology for these functions. For example, the topological relations can be expressed in the terminology of the 9-intersection model (Egenhofer, 1990), RCC-8 (Randell, 1992), or OGC Simple Features (Open Geospatial Consortium, 2011). While implementations of GeoSPARQL do

³ <http://www.opengeospatial.org/standards/gml>

not have to support all of these vocabularies, it is expected that most will. The binary Boolean topological relations can be expressed as either functions in a FILTER clause or as triples between SpatialObjects.

The following is an example of a GeoSPARQL query, which looks for monuments within parks, where both classes are subclasses of Feature. For more detailed examples see the GeoSPARQL specification (Perry, 2010) or (Battle, 2012). (The prefix “geo:” in this example refers to www.opengis.net/ont/geosparql, while the prefix “ex:” refers to an arbitrary example domain.)

```
SELECT ?m ?p
WHERE{
  ?m a ex:Monument ;
     geo:hasGeometry ?mgeo .
  ?p a ex:Park ;
     geo:hasGeometry ?pgeo .
  ?mgeo geo:within ?pgeo .
}
```

3.2 Parliament

Parliament⁴ is an open-source RDF triple store (Figure 1). The outer layers are based on the open-source RDF toolkit Jena⁵, which connects to a novel indexing scheme for RDF triples (Kolas 2009). Parliament provides a SPARQL endpoint for storing and querying RDF triples.

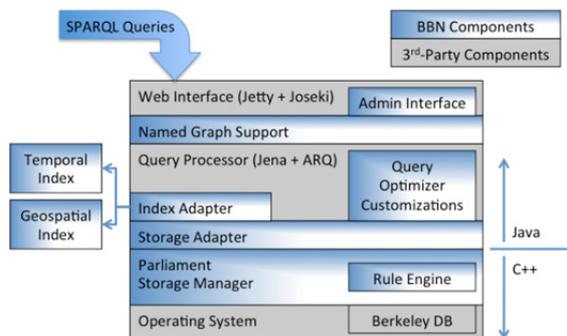


Figure 1: Parliament Architecture.

As the importance of the Geospatial Semantic Web grew, Parliament added spatial and temporal indexing. Initially this was a purely in-memory index designed as a proof of concept. It used OWL-Time⁶ and an ontology based on GeoRSS⁷ as the vocabularies for indexing. More recently the indices

⁴ <http://parliament.semwebcentral.org/>

⁵ <http://jena.apache.org/>

⁶ <http://www.w3.org/TR/owl-time/>

⁷ <http://www.georss.org>

were updated to be persistent. The spatial index can use either an R-tree implementation or an external instance of Postgres. The temporal index uses Berkeley DB. The result is the ability to store and query spatial and temporal RDF efficiently. As the GeoSPARQL standard matured, Parliament was updated to include support for the standard.

4 GeoQuery

GeoQuery is being developed to provide GIS professionals an easy way to explore geospatial semantics in a familiar mapping interface, and to provide the ability to see how the semantics queries are actually built and executed.

The user interface is similar to many web-based mapping sites, where the user can turn layers on and off, and navigate around the map by panning and zooming. The interface also includes the ability to execute two separate queries using pick lists and text boxes, and to perform spatial operations on the results of the two queries. The onscreen areas for performing these operations have separate colors, and the results for each operation are displayed on the map with the same color to make it easy to identify the results for each operation.

Because one of the design goals of GeoQuery was to help explain GeoSPARQL, the full text of each GeoSPARQL query is saved and all of the query text can be viewed at any time.

The GeoSPARQL endpoint we are using for development is an instance of Parliament that has been populated with vector data extracted from the USGS National Map⁸. The data was extracted using the boundaries of the Shenandoah River (Virginia, USA) watershed, and was converted from an ESRI Personal Geodatabase to .N3 files using a custom tool developed by the USGS⁹.

4.1 Design and Operation

Development of GeoQuery began by establishing a set of six general use cases that were gathered by examining example queries from literature. Development followed a prototyping process, the starting point for which was a tool for visualizing GeoSPARQL query results developed by the USGS. The ultimate user interface was developed with the

⁸ See “Products and Services” at <http://nationalmap.gov/index.html>

⁹ USGS NationalMap2RDF conversion tool: http://cegis.usgs.gov/ontology_userguide.html

intention of reproducing some controls and functions commonly found in existing GIS software.

Architecturally, GeoQuery is a web application that uses the web browser as its execution platform (Figure 2). As such, it is written entirely in HTML and JavaScript. The map display functionality is based upon OpenLayers¹⁰, which provides a robust API for obtaining, displaying, and manipulating map tiles. JQuery provides a variety of coding shortcuts that reduce the overall amount of custom JavaScript required. The interface to the GeoSPARQL endpoint is based on Ajax and JSON. All GeoSPARQL queries are generated in JavaScript and then sent via Ajax to the Parliament server. Responses are returned in JSON format.

The current GeoSPARQL endpoint is built with Parliament, but GeoQuery should be compatible with any GeoSPARQL server.

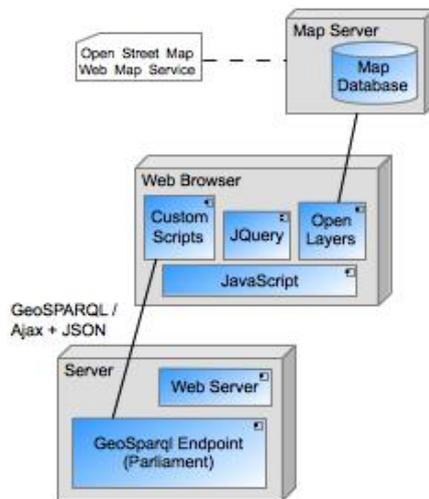


Figure 2: GeoQuery Architecture.

At startup, GeoQuery launches a predefined GeoSPARQL query to request map bounds (which are pre-loaded into the endpoint server) in order to select the initial map display. Other than this initialization, the system is fully event-driven. Each time the user launches a query or executes a spatial operation, the request is translated into GeoSPARQL and delivered via Ajax to the GeoSPARQL endpoint server. Query results, which consist of sets of spatial objects with WKT encoding, are returned as JSON objects, which must be decoded in order to be displayed on the map when appropriate. Some queries also return textual results, which are displayed in pop-up windows. No query

¹⁰ <http://openlayers.org>

optimization is performed in this version, but that would be an obvious next step in development.

4.2 User Queries

The user interface includes a map with basic navigation tools on the right side of the screen and user options for interacting with the data on the left side (Figure 3). The interface allows for two distinct queries to be executed, and for a spatial operation to be performed on the combined results.

For example, finding all of the schools that are present in a particular county (using the USGS National Map data) can be accomplished by defining the two queries and then applying a spatial operation.

First, to find all of the schools, the query can be defined in the Feature 1 query section on the user interface (Figure 3 - query area and results displayed in orange). In the USGS data, the point locations for different types of structures are stored in the class called *structPoint*, with different types of structures coded in a field called *fType*. Schools can be selected by selecting the feature type of *structPoint*, and the feature property *fType* with an *fType* value of 730 (Figure 3, Appendix: Query 1). Then, the Feature 2 query area can be used to select a particular county (query area and results displayed in cyan). Shenandoah County can be selected by choosing the feature type of *countyOrEquivalent*, then selecting to search on the label “Shenandoah” (Query 2). Invoking the Spatial Relationship tool (selection area and results displayed in purple) allows for any of the GeoSPARQL supported spatial operations to be applied to the results of the two searches, such as determining the schools that are within Shenandoah County (Figure 4, Appendix: Query 3).

Queries are created based on predefined patterns, using terms selected by the user from those derived from the data store. GeoQuery does not apply heuristics or make inferences from input, rather it responds in a straightforward way to user selections. Terms used to describe features and their properties are derived from the data store, and GeoQuery does not interpret or modify them. These abilities could be extensions to the tool in future versions.

Providing a complete interface to GeoSPARQL was not a design goal of this project. The set of query forms generated by GeoQuery is a small subset of the (infinite) set of query forms that could be generated in GeoSPARQL. Though more complex query options could be added to GeoQuery to extend the range of resulting queries, it is not

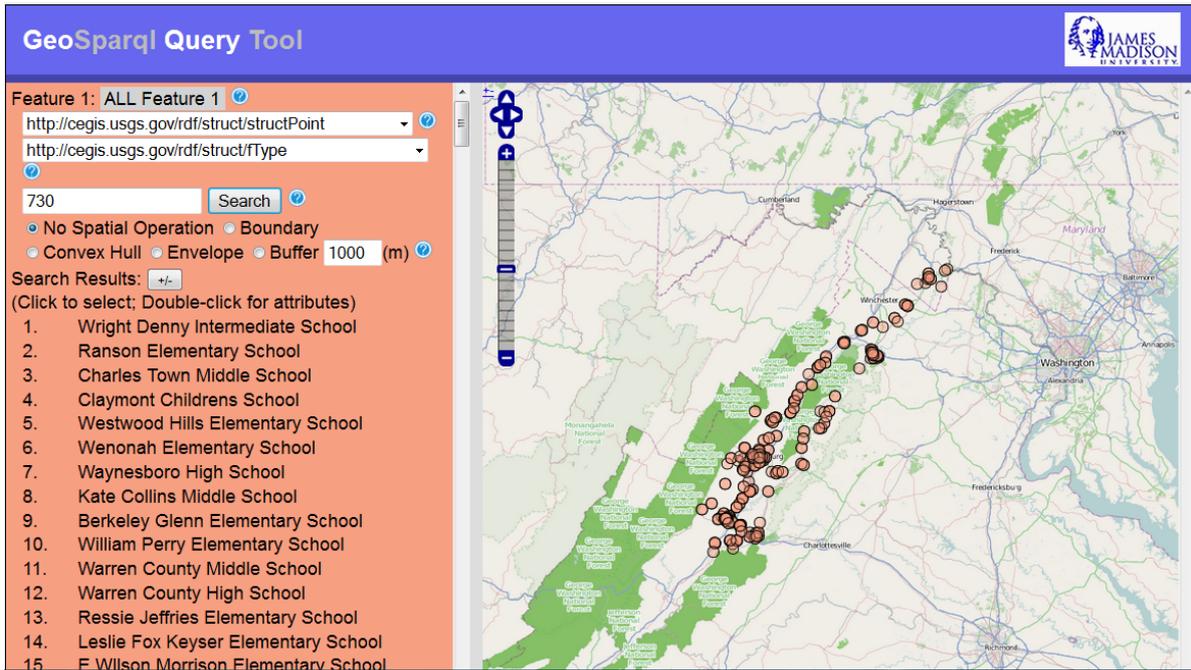


Figure 3: Schools.

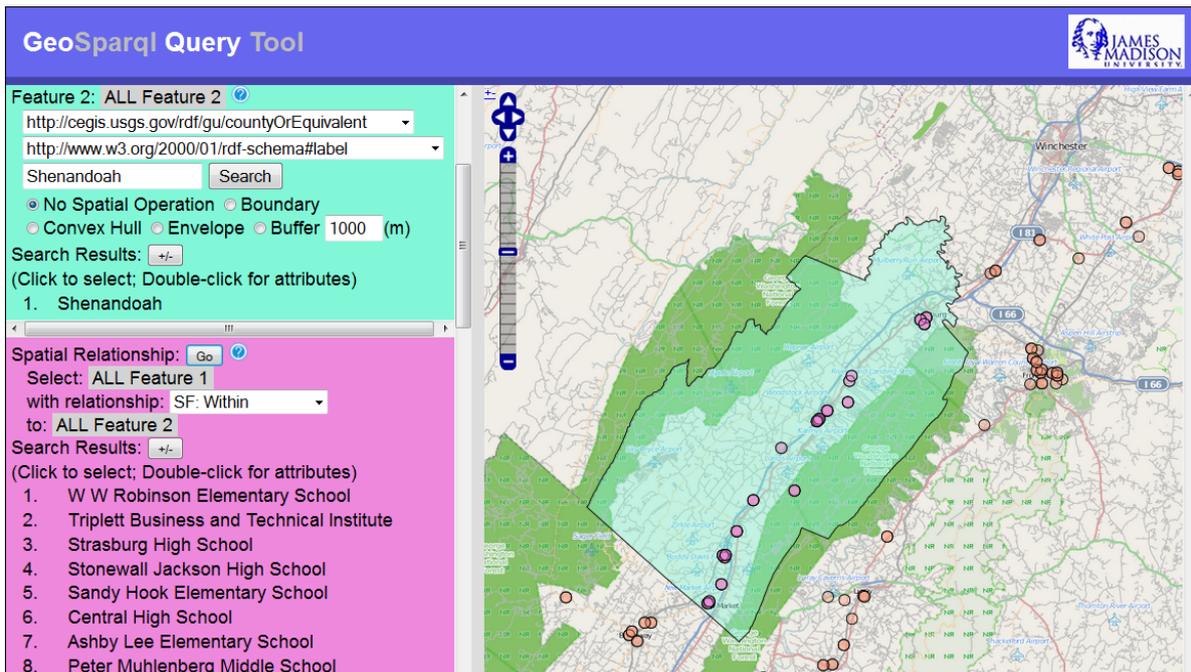


Figure 4: Schools within Shenandoah County.

clear that completeness could be gained without exposing the user to elements of GeoSPARQL syntactic structure, which is counter to the design goals. Instead, we have provided an interface to support commonly used types of queries.

5 CONCLUSIONS

This work illustrates how a geospatial query tool can be successfully implemented based on Semantic

Web technologies such as RDF, SPARQL, and GeoSPARQL. Users can effectively query an RDF geospatial database over the Web, execute spatial operators on the results, and then visualize the results on a map in a familiar format, without knowing a formal query language. This is a first step towards bringing the value of the Semantic Web and open data to geospatial data and users. GeoQuery is an integral step in provided needed query access to the Geospatial Semantic Web.

The current GeoQuery tool is an initial proof of concept. The tool could be improved by replacing USGS National Map URIs and codes with more user-understandable synonyms. We used RDF data directly from The National Map and did not re-code it to be understandable by the general user. This is a limitation of using data directly converted to RDF, but adding definitions or links to ontologies is beyond the scope of this project. The innovation of our work is to take a new paradigm (RDF and GeoSPARQL) and make the data and querying accessible to any Web user using a graphical interface.

Testing to improve the tool could include formal user testing, testing of the tool against other SPARQL endpoints, testing with multiple endpoints simultaneously, and comparing its use against conventional tools. Additional extensions of the tool could include query optimization, the addition of more complex query forms through additional user interface options, and automatic clustering of results. We are also working on methods to automate converting general spatial data to RDF to make more spatial data accessible and to further test the tool.

ACKNOWLEDGEMENTS

A This work was partially supported by the National Science Foundation's Office of Cyberinfrastructure (OCI), INTEROP Grant No. 0955816.

Early versions of GeoQuery were based in part on a prototype query tool developed by USGS

REFERENCES

- Battle, R., Kolas, D., 2012. Enabling the Geospatial Semantic Web with Parliament and GeoSPARQL, *Semantic Web Journal* 3(4): 355-370.
- Bizer, C., Heath, T., and Berners-Lee, T., 2009. Linked Data – The Story So Far, *International Journal on Semantic Web and Information Systems*, Special Issue on Linked Data, 5:1–22.
- Collustra, 2013. (Online). Available: <http://tw.rpi.edu/web/event/TWeD/2013/Fall/Collustra> (30 Sep 2013).
- Das, S., Chong, E. I., Eadon, G., Srinivasan, J., 2004. Supporting Ontology-based Semantic Matching in RDBMS, *Proceedings of the 30th VLDB Conference*, Toronto, Canada, pp. 1054-1065, 2004.
- Denaux, R., Dolbear, C., Hart, G., Dimitrova, V., and Cohn, A., 2011. Supporting Domain Experts to Construct Conceptual ontologies: A Holistic Approach, *Web Semantics: Science, Services and Agents on the World Wide Web*, 9 (2011), pp. 113-127.
- Egenhofer, M. J., and Herring, J. R., 1990. Categorizing binary topological relations between regions, lines, and points in geographic databases. *Technical report, Department of Surveying Engineering*, University of Maine.
- Egenhofer, M. J., 2002. Toward the semantic geospatial web, *Proceedings of the ACM GIS02*, pp. 1-4.
- Garbis, G., Kyzirakos, K., and Koubarakis, M., 2013. Geographica: A Benchmark for Geospatial RDF Stores, accepted at the *2013 International Semantic Web Conference*.
- Geospatial Semantics and Ontology, 2013. Center for Excellence in Geospatial Science, [Online], Available: <http://cegis.usgs.gov/ontology.html> [15 Sep 2013].
- Grandi, F., 2012. Introducing an Annotated Bibliography on Temporal and Evolution Aspects in the Semantic Web, *SIGMOD Record*, December 2012 (Vol. 41, No. 4), pp. 18-21, Available: <http://www-db.deis.unibo.it/~fgrandi/TWbib/TSWbib.html> (15 Sep 2013).
- Hoffart, J., Suchanek, F. M., Berberich, K., and Weikum, G., 2012. YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia (preprint), (Online), Available: <http://www.mpi-inf.mpg.de/yago-naga/yago/publications/aij.pdf> (29 Sep 2013).
- Kolas, D., 2009. Supporting Spatial Semantics with SPARQL, *Transactions in GIS, Vol. 12*, Issue s1, pp. 5-18, December 2008.
- Koubarakis, M., Karpathiotakis, M., Kyzirakos, K., Nikolaou, C, and Sioutis, M., 2012. Data Models and Query Languages for Linked Geospatial Data, *Reasoning Web 2012*, LNCS 7487, T. Eiter and T. Krennwallner (Eds.) pp. 220-328, 2012.
- Kyzirakos, K., Karpathiotakis, M., and Koubarakis, M., 2012. Strabon: A Semantic Geospatial DBMS, *Proceedings of the 11th International Semantic Web Conference (ISWC 2012)*, Boston, USA, November 11th-15th.
- The Linked Data Web, (Online), Available: <http://www.ordnancesurvey.co.uk/education-research/research/linked-data-web.html> (16 Sep 2013).
- Ma, L., Wang, C., Lu, J., Cao, F., Pan, Y., Yu, Y., 2008. Effective and Efficient Semantic Web Data Management over DB2, *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pp. 1183-1194.
- Manola, F., Miller, E. (Ed.s), 2004. RDF Primer, (Online), Available: <http://www.w3.org/TR/rdf-primer> (25 Dec 2013).

Open Geospatial Consortium, 2011. OpenGIS Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture, (Online), Available: <http://www.opengeospatial.org/standards/sfa> (25 Dec 2013).

Perry, M., Herring, J. (Eds.), 2010. OGC GeoSPARQL - A Geographic Query Language for RDF Data, Open Geospatial Consortium, (Online), Available: <http://www.opengis.net/doc/IS/geosparql/1.0> (16 Sep 2013).

Prud'hommeaux, E., Seaborne, A. (Eds.), 2008. SPARQL Query Language for RDF, (Online), Available: <http://www.w3.org/TR/rdf-sparql-query> (25 Dec 2013).

Randell, D. A., Cui, Z., and Cohn, A. G., 1992. A spatial logic based on regions and connection. In *Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning*.

Terra Cognita, 2012. (Online), Available: <http://iswc2012.semanticweb.org/workshops/TerraCognita.html> (15 Sep 2013).

Varanka, D. (Ed.), 2012. *Introduction to Geospatial Semantics and Technology Workshop Handbook*, 2012 University Consortium for Geographic Information Science Symposium, (Online), Available: <http://pubs.usgs.gov/of/2012/1109/> (15 Sep 2013).

APPENDIX

Query 1: select structPoint with fType=730

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?feature ?label
WHERE {
  # Select features of the specified type:
  ?feature rdf:type <http://cegis.usgs.gov/rdf/struct/structPoint> .
  ?feature rdfs:label ?label .
  # Filter features by property:
  ?feature <http://cegis.usgs.gov/rdf/struct/fType> ?obj .
  FILTER( regex(str(?obj), "730", "i" ) ) .
  # Eliminate the group of features ending in "/None"
  FILTER(! regex(str(?feature), "/None$", "i" ) ) .
}
.....Not showing individual queries to obtain geometry and
map the features....
```

Query 2: Select & draw Shenandoah from countyOrEquivalent

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?feature ?label
WHERE {
  # Select features of the specified type:
  ?feature rdf:type
<http://cegis.usgs.gov/rdf/gu/countyOrEquivalent> .
  ?feature rdfs:label ?label .
  # Filter features by property:
  ?feature <http://www.w3.org/2000/01/rdf-schema#label> ?obj .
  FILTER( regex(str(?obj), "Shenandoah", "i" ) ) .
  # Eliminate the group of features ending in "/None"
  FILTER(! regex(str(?feature), "/None$", "i" ) ) .
```

```
}
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
SELECT ?wkt
WHERE {
  <http://cegis.usgs.gov/rdf/gu/Features/1673918>
  geo:hasGeometry ?g .
  ?g geo:asWKT ?wkt .
}
```

Query 3: Schools within Shenandoah County

```
PREFIX geo: <http://www.opengis.net/ont/geosparql#>
PREFIX geof: <http://www.opengis.net/def/function/geosparql/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX units: <http://www.opengis.net/def/uom/OGC/1.0/>
SELECT DISTINCT ?feature ?label
WHERE {
  # Feature 1:
  # Select features of the specified type:
  ?feature rdf:type <http://cegis.usgs.gov/rdf/struct/structPoint> .
  ?feature rdfs:label ?label .
  # Filter features by property:
  ?feature <http://cegis.usgs.gov/rdf/struct/fType> ?obj1 .
  FILTER( regex(str(?obj1), "730", "i" ) ) .
  # Eliminate the group of features ending in "/None"
  FILTER(! regex(str(?feature), "/None$", "i" ) ) .
  ?feature geo:hasGeometry ?g1 .
  ?g1 geo:asWKT ?wkt1 .
  # Feature 2:
  # Select features of the specified type:
  ?feature2 rdf:type
<http://cegis.usgs.gov/rdf/gu/countyOrEquivalent> .
  # Filter features by property:
  ?feature2 <http://www.w3.org/2000/01/rdf-schema#label> ?obj2 .
  FILTER( regex(str(?obj2), "Shenandoah", "i" ) ) .
  # Eliminate the group of features ending in "/None"
  FILTER(! regex(str(?feature2), "/None$", "i" ) ) .
  ?feature2 geo:hasGeometry ?g2 .
  ?g2 geo:asWKT ?wkt2 .

  # spatial relationship
  FILTER (geof:sfWithin(?wkt1, ?wkt2)) .
}
```

(Note: Some of the queries contain a filter term intended to eliminate features ending in “/None”. These are features that have incomplete definitions, an anomaly of the test dataset that was used.)

SIWAM: Using Social Data to Semantically Assess the Difficulties in Mountain Activities

Javier Rincón Borobia, Carlos Bobed, Angel Luis Garrido and Eduardo Mena

*IIS Department, University of Zaragoza, Zaragoza, Spain
javirbh@gmail.com, {cbobed, garrido, emena}@unizar.es*

Keywords: Semantic Web, Information Extraction, Ontologies, Social Network.

Abstract: In the last few years, the amount of people moving to the mountains to do several activities such as hiking, climbing or mountaineering, is steadily increasing. Not surprisingly, this has come along with a raise in the amount of accidents, which are mainly due to the inexperience of the people, and the lack of information and proper planning. Although one could expect to find appropriate updated information about this issue on the Internet, most of the information related to mountain activities is stored in personal blogs, or in Web sites that are not exploiting the possibilities that the Semantic Web and the Social Web offer regarding content generation and information processing.

In this paper, we present SIWAM, a semantic framework oriented to share and evaluate the difficulties of mountain activities. It provides a thematic social network front-end to enable users to share their descriptions about their own experiences. Using text mining techniques on these descriptions, it extracts relevant facts about these experiences, which are used to evaluate the difficulty of the particular activity. The evaluation is done according to a well-established standard for evaluating the difficulty of mountain activities (MIDE), which is modeled in the system using ontologies.

1 INTRODUCTION

Mountain activities comprise a set of sports that are amongst the most practiced in the world. The amount of people practising them is increasing year by year all around the world (Global Industry Analyst, Inc., 2012; Jenkins, 2013). This steady increment of practitioners along with the fact that mountain activities are considered extreme sports make the security be an important and recurring matter of study. There are research areas (specially medical ones) (Chamarro and Fernández-Castro, 2009), and organizations, such as the International Mountaineering and Climbing Federation, which are specially concerned about mountain accidents (UIAA Mountaineering Commission, 2004). However, in spite of all the efforts, the amount of accidents is, not surprisingly, increasing as more and more people move to the mountains. From the yearly reports of the rescue groups in Spain (Ministerio del Interior, 2012), we can point out that the main problems regarding accidents are the inexperience of the people, and the lack of information and proper planning.

While mountain guide books can be a source of information about the tracks and the environment where

the activity is held, it seems pretty safe to assume that, as the main cause of accidents is lack of information, people are not using them to plan their activity (when even they plan it). Instead, in these times, people go to the Internet to look for information as fast as possible, regardless the possible security implications. In the case of mountain activities, this information is mainly available as descriptions of the different experiences in text format (e.g., blog entries).

However, trusting these on-line descriptions might be a double-edged sword: On the one hand, they might be really useful as they hold information about the activity; but, on the other hand, they are describing unique experiences whose setup and conditions might not be applicable to other situations. A climbing route might have many variations, shortcuts, forks, etc., within the same path. In a mountain environment, a slight detour might turn an easy track into a challenging one. Besides, the weather conditions also affect strongly to the mismatch between the descriptions and the track difficulties. As an example, in high mountain, the orientation is mainly guided by milestones that may be covered by snow during winter season.

Moreover, the level of expertise of the person describing the activity might lead to dangerous biases,

as, for an experienced mountaineer, several actions might be considered too easy to be worthy of being described, although they may suppose a danger for an amateur practitioner. Therefore, there is a need for methods to support the correct evaluation of the different activities taking into account the expertise and physical conditions of the practitioner.

Currently, in the mountain Web sites, the main method to introduce information about an activity is form-oriented, with several predefined fields frequently including a textual one, where users can provide a more detailed description of their experiences. In fact, the information that can make a difference may be hidden in these textual descriptions (e.g., “the track is quite easy, but there is a fork that might lead to a . . .”). An improvement could be to include more detailed forms; however, the amount of details to be included might be overwhelming for the users, resulting in a decrease of user collaboration.

In this paper, we present SIWAM¹, a system that provides a semantic framework to improve users’ information in the domain of mountain activities with the main goal of improving their security. It consists of three main modules: 1) a social Web front-end (RSAM), where users can share their experiences and provide information about their profiles; 2) a text-extraction module (MECMIDE) that is in charge of detecting and extracting the relevant facts from the activity descriptions provided by the users; and, finally, 3) an activity semantic evaluator that, given the relevant facts, classifies the different activities according to a well-defined mountain activities standard (MIDE), which is modeled in the system by an ontology network. The complete set of individual evaluations can be taken into account to provide a more accurate evaluation of the different activities depending on different aspects such as the weather conditions, the season, the user’s profile, and so on.

The rest of the paper is as follows. In Section 2, we present the architecture of our system, detailing the main modules. In Section 3, we introduce the system’s ontology and the MIDE standard (Roche, 2002), and how we have modeled the latter to be exploited by SIWAM. Section 4 and Section 5 detail the information extraction module (MECMIDE) and the evaluation module (VALMIDE), respectively. A complete example of how SIWAM works is presented in Section 6. We discuss some related work in Section 7. Finally, the conclusions and future work are drawn in Section 8.

¹SIWAM stands for *Sistema de Información Web para Actividades de Montaña* (in Spanish, which means Mountain Activities Web Information System).

2 ARCHITECTURE OF THE SYSTEM

In this section, we present a general overview of SIWAM, describing its aim and general architecture. SIWAM is conceived to be a social site where users of all expertise share their experiences performing different mountain activities. Using text-mining and Semantic Web techniques, SIWAM is capable of processing this information to assess the difficulty of the different tracks and activities according to a well-established mountain activities standard. To do so, SIWAM exploits the knowledge stored in the System Ontology (see Figure 1). In this way, SIWAM offers much more precise and updated information about the different activities, having as a final objective to improve the mountaineers security (no matter their experience levels).

As it can be seen in Figure 1, SIWAM consists of three main modules:

- **RSAM (Social Network for Mountain Activities:)** This is the Web front-end of SIWAM, and supports the common features of a social site (e.g., user profiles, instant messaging, groups, contacts, etc.). Its functionality is specifically extended with features oriented to the management and sharing of information about mountain activities (e.g., including new ones, adding descriptions and experiences, sharing maps, GPS routes, etc.). SIWAM processes these shared descriptions and experiences in background to obtain the information about the safeness of a particular activity.
- **MECMIDE (Concept Extraction Module:)** It is the module in charge of processing each of the descriptions and extracting the relevant facts out of them. To do so, it uses Freeling (Carreras et al., 2004) to analyze the texts, looking for relevant patterns which might contain information. These patterns are mapped to different concepts and properties of the System Ontology, so MECMIDE outputs a set of axioms in the form of RDF² triples which are correctly aligned with this ontology.
- **VALMIDE (Evaluator Module:)** This module takes as input the facts that MECMIDE has extracted from the text, and, with the help of a Description Logics reasoner (Baader et al., 2003) (DL reasoner from now on), evaluates the difficulty of the activity according to the System Ontology. This ontology models the MIDE (Roche, 2002) standard to evaluate the difficulty of mountain activities regarding several criteria. The

²<http://www.w3.org/TR/rdf-primer/>

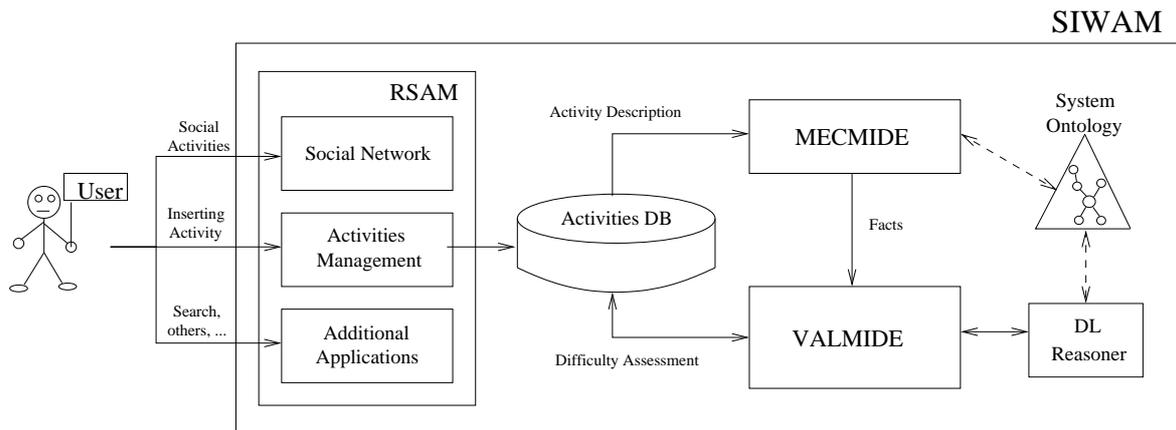


Figure 1: Architecture of SIWAM.

global evaluation of a particular activity is obtained combining the single evaluations of all the people that have done it. Considering all the descriptions makes the evaluation more robust against omitted information.

In the following, we will focus on the semantic part of SIWAM. First, we present how we have modeled the MIDE evaluation standard in the System Ontology. Then, we overview how MECMIDE extracts the information from the texts using the vocabulary defined in the used ontology. Finally, we explain how this knowledge is used by VALMIDE to assess the difficulty of a particular activity.

3 SYSTEM ONTOLOGY

The System Ontology stores all the information needed to perform the evaluation of the different activities. Following the directives given in NeON Methodology (Suárez-Figueroa, 2012), we have built an ontology network to make it possible to follow a modular development. In Figure 2, we can see the inner structure of this ontology:

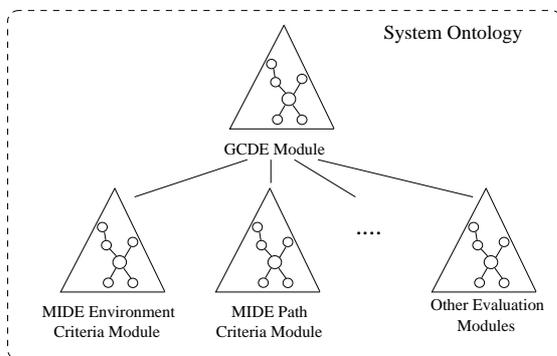


Figure 2: Inner structure of the ontology used by SIWAM.

- The Global Criteria Excursions Definition (GCDE) ontology module integrates the different modules that model different evaluation criteria, and stores information about how they must be used to evaluate an activity. In particular, for each of the activity types that SIWAM handles, GCDE stores the evaluation modules that are applicable to it and the methods to be used to do so. This enables the system to react automatically to the addition of new evaluation modules and methods.

Currently, it only contains the description of one method of evaluation, which is the one applied to the MIDE evaluations (we will see it later in Section 5); however, it is interesting to specify its role in the system as it provides our system with a flexible method to add/remove evaluation modules.

- For each evaluation criteria to be used, there is an ontology module that models it in the system. In particular, we have selected the MIDE criteria for SIWAM, as it is a well-established standard for mountain activities evaluation.

The motivation for using ontologies to model the different evaluation criteria is two-fold: On the one hand, so far, there has not been not any initiative to model the domain of mountain activities and provide a shared vocabulary; on the other hand, using ontologies provides us with a logical framework to perform the activity evaluation in a more tractable and manageable way than using pure rule systems, as we will see in Section 5. Moreover, the use of an ontological model as a base for the text information extraction has been successfully applied in many works such as (Garrido et al., 2012; Vogrincic and Bosnic, 2011; Garrido et al., 2013; Kara et al., 2012).

Thus, in the rest of the section, we focus on how we have modeled the evaluation criteria knowledge in this ontology. To do so, we firstly overview the MIDE

standard, and then we detail how we have captured this information in the appropriate ontology module.

3.1 MIDE Evaluation Standard

The MIDE standard (Roche, 2002) is focused on the evaluation of mountain tracks. To do so, MIDE proposes to tag a particular excursion (*experiences* using the SIWAM terminology) with two different kinds of information: 1) information about the track (name, type, accumulated height difference, etc.) and the conditions of the actual activity performing (hour, season, weather conditions, etc.); and 2) evaluation information, which provides difficulty values ranged in 1 – 5 for different aspects of the track, namely *Environment*, *Path*, *Journey*, and *Physical Effort*.

Most of the first kind of information can be captured using simple forms (e.g., times, height differences, distances, etc.). For the second type of information, MIDE provides guidelines to evaluate each of those values. While the Physical Effort can be obtained using different formulae, the rest of values (Environment, Path and Journey) are assessed using a set of criteria that might or might not be present in the track. Then, depending on the number of criteria the activity fulfils, it is assigned a difficulty value from 1 to 5 for that particular aspect.

Capturing these criteria using formularies is more complicated. They would require complex forms, which would be long and time-consuming to fill, a task that not all the users are prone to do. Fortunately, this is the kind of information that is usually comprised in the textual descriptions of the user’s activities. Thus, the first step to perform an automatic evaluation of the track is to model these criteria to be used by SIWAM. In the following subsection, we present our approach to do this task.

3.2 Modeling MIDE with Ontologies

The development of the ontology modules that model the MIDE evaluation criteria was carried out in two well differentiated stages:

1. Modeling the mountain domain: The objective of this first stage was to obtain and organize the elements (concepts and properties) within the domain of the mountain activities. We did not aim at modeling the whole domain at once, but incrementally, taking each of the MIDE evaluation criteria as the competence questions (Grüniger and Fox, 1994) for each iteration. This way, we could assure that we had modeled all the elements needed to model the MIDE knowledge.
2. Modeling the criteria: In the case of MIDE, the evaluation is performed by checking whether a particular activity fulfils or not an specific (and complex) condition. From an ontological point of view, this is equivalent to say that the activity belongs to a particular type of activities defined by this condition. Thus, we used *defined concepts* to model them. In Description Logics (Baader et al., 2003), the underlying formalism of OWL³, a defined concept provides a complete definition of its members, that is, it establishes *necessary and sufficient* conditions for an instance to belong to it. This kind of definitions enables DL reasoners to classify the instances according to them. In the following section, we will see how VALMIDE module takes advantage of this issue.

Moreover, we added also the information about how many criteria the activities must fulfil to be assigned each of the 1 – 5 values. We modeled this information also as definitions, but they do not affect to the reasoning process. They are just consulted by VALMIDE to obtain the mapping between the number of criteria and the final value.

We now present two different examples to illustrate how these definitions comprise the knowledge about the criteria. We focus on the subdomain of the Environment module, this is, criteria about the harshness of the environment:

 - Criteria “crossing a place farther than 1 hour (walking time) from a inhabited place” is modeled as a new concept whose definition is


```
(actionPerformed some CrossRemoteArea3H)
```

 along with the following definitions


```
CrossRemoteArea3H equivalentTo
(Cross
and (actionPerformedIn some RemoteArea3H))

RemoteArea3H equivalentTo
(SingularElement
and (distantFrom some InhabitedPlace)
and (isWalkingDistance some integer[>=3]))
```
 - Criteria “high probability of temperatures under 0°C” and “high probability of temperatures under 10°C” are modeled respectively as


```
(hasMinTemperature some integer[<0])
```

 and


```
(hasMinTemperature some integer[<-10])
```

³<http://www.w3.org/TR/owl-primer/>

These definitions enable VALMIDE to classify the texts with the help of a DL reasoner. In the following section, we present the MECMIDE module, that enables SIWAM to obtain the facts to be classified out from the text.

4 MECMIDE MODULE

The MECMIDE module is in charge of extracting the relevant information out from the descriptions provided by the users. It works as follows: For each fact or concept in the System Ontology that we want to be able to detect, MECMIDE has a list of search patterns that are to be looked up in the texts. For instance, the concept “Use hands on step” has a list of patterns such as “utilize hand”, “use hand”, “require hand use”, etc. That list contains the lemmatized form of each word used. The patterns do not include articles, conjunctions, prepositions, or other words that lack of intrinsic semantic value.

To enrich the pattern-search analysis, MECMIDE includes two additional advanced semantic features:

- The processing window: MECMIDE searches each of the words composing the patterns in an area defined according to a given number of words. This feature enables MECMIDE to detect a pattern in the text regardless the word order. This is useful because sentences can hold different structures with the same words⁴.
- The use of synonyms: Instead of looking just for the exact words of each pattern, MECMIDE considers also their synonyms. For example the word “way” has the same meaning as “traverse”, “track”, “trail”, “path”, “route”, or “itinerary”. The use of synonyms is solved using a lexical database, like WordNet (Miller, 1995), which allows MECMIDE to transform the words that integrate each pattern in *synsets*, i.e., the canonical form of its meaning. Then, the search is carried out using these synset, which broadens the vocabulary coverage of the patterns.

In many cases, it is not sufficient that a single pattern is recognized to deduce that the text is related to a particular concept. So, on top of these text patterns, MECMIDE uses a set of rules to decide if we can actually associate a text with a fact or concept in the VALMIDE concept (e.g., the need of a minimum number of patterns to deduce whether a particular description is related to an ontology concept). Thus, the

⁴This structure richness is very typical when processing Spanish texts.

process followed by MECMIDE to process each text is composed by three steps:

1. Lemmatization of the texts: Freeling (Carreras et al., 2004) is used as a tagger and lemmatizer, to filter stop words and to obtain the lemma of each word of the text, respectively.
2. Looking for patterns: To do this, the words that form the patterns are converted into *synsets* to improve the quality of searches. For example, “dangerous way” is equivalent to “slippery path”. In our current implementation, we have chosen EuroWordNet (Vossen, 1998) as our lexical database because our prototype is in Spanish.
3. Applying the rules: The set of rules is evaluated to see whether the text must be linked to a concept, i.e., whether a fact can be derived from the text.

The result of this analysis is a set of concepts and facts that are associated to the text. In the following section, we present how this set of facts is used by VALMIDE to evaluate the activity according to the knowledge stored in the System Ontology.

5 VALMIDE MODULE

As we have seen in Section 3, the ontology used by SIWAM is an ontology network composed by two levels: One with information about the evaluation methods used for each evaluation criteria (GCDE Module), and the other with the actual knowledge needed to perform the actual evaluation (criteria modules). VALMIDE uses this information to perform the evaluation of each single activity. This process is composed by the following steps:

1. Deciding the evaluation modules: When it receives the description text along with the extracted facts, VALMIDE consults the GCDE module to see which evaluation modules are applicable to the type of activity that is being described.
2. Obtaining the evaluation methods: GCDE contains information about the specific method to be used for each evaluation module. VALMIDE consults it to know how to handle a specific module. This way, we can attach evaluation modules and methods in a flexible and decoupled manner.
3. Evaluating the activity description: For each module, VALMIDE asserts the facts in a copy of the evaluation module and classifies it. The classification and inferring capabilities of DL reasoners enable VALMIDE to make it explicit knowledge that otherwise would remain implicit. This information is used to calculate the actual evaluation.

4. Combining evaluations: Depending on the evaluation module, VALMIDE can consider whether to just evaluate the activity as a single one, or to combine previous evaluations to achieve an agreed evaluation. The information of the method to use is stored in the GCDE module.

Regarding the MIDE Modules, we have developed an evaluation method based on the activity description classifications. As we have explained in Section 3, each MIDE criteria is modeled as a defined concept. The MIDE evaluation method obtains the agreed number of criteria fulfilled using the following formulae:

$$MIDE\ value(A_{instance}) = \sum_{i=1}^{|Crit|} fulfil(crit_i)$$

with

$$fulfil(crit_i) = \begin{cases} 1 & \text{if } \frac{\sum_{k=1}^{|Desc|} crit_i(desc_k)}{|Desc|} > CT \\ 0 & \text{otherwise} \end{cases}$$

where $A_{instance}$ is the activity we are evaluating, $Crit$ is the set of criteria to be evaluated (the set of concept definitions), $Desc$ is the set of individual descriptions for $A_{instance}$, CT is a confidence threshold in $0..1$, and $crit_i(desc_k)$ is a function that evaluates whether a description is an instance of a particular criteria (this function is evaluated with the help of the DL reasoner, and returns 1 if the belonging relationship is entailed).

The above method counts how many of the criteria the activity fulfils by asking the DL reasoner whether the activity description belongs to each of the criteria definition concepts. Then, the MIDE information for each particular activity (the evaluation of each of its descriptions) is combined to establish whether a particular criterion is met. This is done by calculating whether it is present in the descriptions in a percentage above a particular confidence threshold. Finally, SIWAM translates the agreed number of criteria into the MIDE final value (ranged in 1 – 5 values) by consulting the information also stored in the module.

In the following section, we present two excerpts of an actual description to illustrate each of the steps that our system takes to process the information from text to the actual evaluation.

6 COMPLETE EXAMPLE

Although our system is in its early stages of implementation, we have already carried out some concept proofs that support the approach. In particular, we have used descriptions that correspond to real experiences of four different mountain ascensions located in

the Pyrenees, a range of mountains that forms a natural border between France and Spain. The original texts are in Spanish, but we have translated the interesting excerpts used in this section to illustrate how SIWAM works.

In the ascension to the Petit Vignemale (3032 m), one of the mountaineers wrote:

“... It took us a little more than 3 hours to reach the Refuge of Oulettes de Gaube, which was closed (we already knew it). This refuge is located at a height of 2151 m., in an isolated high mountain place, ...

... The temperature at such height is -18°C , with a 20 km/h wind, ...”

From these excerpts (*ex1*), MECMIDE extracts the following facts:

- From the subject of the description (is given by RSAM):

```
ex1 isA Hike
```

- From the location excerpt:

```
refugeOulettesGaube isA InhabitedPlace
```

```
zone2 isA SingularElement
zone2 distantFrom refugeOulettesGaube
zone2 isWalkingTime 3
```

```
crossZone2 isA Cross
crossZone2 actionPerformedIn zone2
```

```
ex1 actionPerformed crossZone2
```

where *zone2* is the area which the mountaineers were traversing, and *crossZone2* is the action of traversing it.

- From the temperature excerpt:

```
ex1 hasMinTemperature -18
```

The instances created in the extraction are related to the activity instance, as MECMIDE assumes that a text contains the description of a single activity. Thus, VALMIDE asserts these axioms in the Environment module, and, with the help of a DL reasoner, infers the following (recall the definitions for the criteria presented in Section 3):

- From the location axioms:

1. *zone2* is a *RemoteZone3H* as it fulfils that is a *SingularElement* and is distant an *InhabitedPlace* (*refugeOulettesGaube*), and is at a walking distance of more than 3 hours.
2. *crossZone2* is a *CrossRemoteZone3H* as it is a *Cross* action performed in a *RemoteZone3H*.

3. *ex1* is an instance of the concept definition associated to the MIDE criterion of crossing a remote area at 3 hours (walking distance) as it has an action that is a *CrossRemoteZone3H*.
- From the temperature axioms:
 1. *ex1* is an instance of the concept definition associated to the criterion about temperatures below -10°C .
 2. *ex1* is also an instance of the concept definition associated to the criterion about temperatures below 0°C .

VALMIDE thus calculates the final evaluation counting the criteria that *ex1* is instance of⁵. Note how *ex1* is added two points in the MIDE criteria due to its temperature as it fulfils two different criteria (below 0°C and below 10°C). This is coherent with the way the MIDE standard evaluates the activity.

7 RELATED WORK

The spread and presence of mountain activities thematic sites on the Web have been quite low compared to other fields. To the best of our knowledge, there is no such an approach as ours in any of the current Web sites about mountain activities.

Regarding the System Ontology, we have not found any other ontology modeling the domain of mountain activities. The closest works are related to the tourism domain (Fodor and Werthner, 2005; Prantner et al., 2007; Barta et al., 2009; Mouhim et al., 2011), although they have different aims as ours as they are mainly oriented to model the tourism domain within the context of the e-commerce. Anyway, we have considered all of them to capture the vocabulary of our ontology.

The most related works to SIWAM can be found in the Information Extraction (IE) field. In particular, according to (Wimalasuriya and D., 2010), SIWAM can be classified as an Ontology-Based Information Extraction (OBIE) system, which is extended with the reasoning capabilities of VALMIDE module. In this context, there are several approaches oriented to automatic content annotation (Cimiano et al., 2004; Buitelaar et al., 2008). PANKOW (Cimiano et al., 2004) processes Web pages looking for instances of a given ontology, thus automatically annotating the Web page with metadata about its content. SOBA (Buitelaar et al., 2008) is oriented to obtain structured information out from semi-structured resources (populate a

⁵In this example, we are considering just one description so the formulae in the previous section is reduced just to count the criteria concepts which *ex1* belongs to.

knowledge base). However, these systems aim only at annotation and fact extraction, while SIWAM uses this extracted information to perform the evaluation of the different activities exploiting the model in the ontology and the DL reasoner classifying capabilities.

Without leaving the Information Extraction field, it is worthy mentioning several approaches (Wu et al., 2008; Cimiano and Völker, 2005) that have as its main goal to construct the ontology that is behind the processed information. For example, Kylin (Wu et al., 2008) uses extraction techniques against Wikipedia's articles to obtain a structured schema out from them. It uses also WordNet along with *machine learning* techniques to obtain the final ontology. However, constructing the ontology is not the objective of SIWAM. SIWAM exploits the domain model to detect important facts in the extraction stage, and then, the model is used to evaluate different aspects of each of the input activities (via their descriptions).

8 CONCLUSIONS AND FUTURE WORK

In this paper, we have presented SIWAM, a complete framework to share information about mountain activities. Apart from its social network features, SIWAM extracts and infers new information from the descriptions provided by the users to help assessing the difficulties of the different activities. This is done to improve the information for the practitioners in order to reduce the risks in the mountains. Moreover, to the best of our knowledge, we have developed the first ontology aimed at modeling mountain activities⁶, and at modeling a standard evaluation method such as MIDE. Our system has the following features:

- It uses the descriptions provided by the users to extract relevant facts aligned to an ontology. This is a source of information which was almost unexploited in the mountain domain.
- It uses the extracted information to evaluate the agreed difficulty of each activity with the help of a DL reasoner. To do so, it exploits the inferring capabilities of the reasoner, along with the evaluation criteria definitions.
- It is completely ontology guided: the adoption of a modular evaluation method makes it possible to extend SIWAM with different evaluation standards in a flexible and efficient way.

⁶We cannot make it available by the time we are writing the paper due to several project restrictions.

Currently, the prototype is under development, although the preliminary results are very promising. As future work, apart from testing the user behaviour, we want to extend the modeled activities and include further evaluation methods taken from the field of recommender systems.

ACKNOWLEDGEMENTS

This research work has been supported by the CICYT project TIN2010-21387-C02-02 and DGA-FSE.

REFERENCES

- Baader, F., Calvanese, D., McGuinness, D., Nardi, D., and Pastel-Schneider, P. (2003). *The Description Logic Handbook. Theory, Implementation and Applications*. Cambridge University Press.
- Barta, R., Feilmayr, C., Pröll, B., Grün, C., and Werthner, H. (2009). Covering the semantic space of tourism: An approach based on modularized ontologies. In *Proc. of the 1st Workshop on Context, Information and Ontologies (CIAO'09), Heraklion (Greece)*, pages 1–8. ACM.
- Buitelaar, P., Cimiano, P., Frank, A., Hartung, M., and Racioppa, S. (2008). Ontology-based information extraction and integration from heterogeneous data sources. *International Journal of Human-Computer Studies*, 66(11):759–788.
- Carreras, X., Chao, I., Padró, L., and Padró, M. (2004). FreeLing: An open-source suite of language analyzers. In *Proc. of the 4th Intl. Conf. on Language Resources and Evaluation (LREC'04)*, pages 239–242. European Language Resources Association.
- Chamarro, A. and Fernández-Castro, J. (2009). The perception of causes of accidents in mountain sports: A study based on the experiences of victims. *Accident Analysis & Prevention*, 41(1):197–201.
- Cimiano, P., Handschuh, S., and Staab, S. (2004). Towards the self-annotating web. In *Proc. of the 13th Intl. Conf. on World Wide Web (WWW'04), New York (NY, USA)*, pages 462–471. ACM.
- Cimiano, P. and Völker, J. (2005). Text2Onto: A framework for ontology learning and data-driven change discovery. In *Proc. of the 10th Intl. Conf. on Natural Language Processing and Information Systems (NLDB'05), Alicante (Spain)*, pages 227–238. Springer Verlag.
- Fodor, O. and Werthner, H. (2005). Harmonise: A step toward an interoperable e-tourism marketplace. *International Journal of Electronic Commerce*, 9(2):11–39.
- Garrido, A. L., Buey, M. G., Ilarri, S., and Mena, E. (2013). GEO-NASS: A semantic tagging experience from geographical data on the media. In *Proc. of the 17th East-European Conf. on Advances in Databases and Information Systems (ADBIS'13), Genoa (Italy)*, pages 56–69. Springer Verlag.
- Garrido, A. L., Gómez, O., Ilarri, S., and Mena, E. (2012). An experience developing a semantic annotation system in a media group. In *Proc. of the 17th Intl. Conf. on Natural Language Processing to Information Systems (NLDB'12), Groningen (The Netherlands)*, pages 333–338. Springer Verlag.
- Global Industry Analyst, Inc. (2012). Extreme sports: A global industry outlook.
- Grüniger, M. and Fox, M. S. (1994). The role of competency questions in enterprise engineering. In *Proc. of the IFIP WG5.7 Workshop on Benchmarking - Theory and Practice, Trondheim (Norway)*, pages 22–31. Springer.
- Jenkins, M. (2013). Maxed out on Everest. *National Geographic*, 32(6):94–113.
- Kara, S., Alan, Ö., Sabuncu, O., Akpınar, S., Cicekli, N. K., and Alpaslan, F. N. (2012). An ontology-based retrieval system using semantic indexing. *Information Systems*, 37(4):294–305.
- Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.
- Ministerio del Interior (2012). Anuario estadístico del Ministerio del Interior. <http://www.interior.gob.es/file/62/62261/62261.pdf>, accessed September 13, 2013.
- Mouhim, S., Aoufi, A. E., Cherkaoui, C. E., Douzi, H., and Mammas, D. (2011). A knowledge management approach based on ontologies: The case of tourism. *International Journal of Computer Science & Emerging Technologies*, 2(6):362–369.
- Prantner, K., Ding, Y., Luger, M., Yan, Z., and Herzog, C. (2007). Tourism ontology and semantic management system: State-of-the-arts analysis. In *Proc. of IADIS Intl. Conf. WWW/Internet 2007, Vila Real (Portugal)*, pages 111–115. IADIS Press.
- Roche, A. P. (2002). *MIDE: Método de Información de Excursiones*. Federación Aragonesa de Montañismo. <http://www.montanasegura.com/MIDE/manualMIDE.pdf>, accessed September 13, 2013.
- Suárez-Figueroa, M. C. (2012). *NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse*. IOS Press.
- UIAA Mountaineering Commission (2004). *To Bolt or not to Be*.
- Vogrincic, S. and Bosnic, Z. (2011). Ontology-based multi-label classification of economic articles. *Computer Science and Information Systems*, 8(1):101–119.
- Vossen, P. (1998). *EuroWordNet: a multilingual database with lexical semantic networks*. Kluwer Academic Boston.
- Wimalasuriya, D. C. and D., D. (2010). Ontology-based information extraction: An introduction and a survey of current approaches. *Journal of Information Science*, 36(3):306–323.
- Wu, F., Hoffmann, R., and Weld, D. S. (2008). Information extraction from Wikipedia: Moving down the long tail. In *Proc. of the 14th ACM Intl. Conf. on Knowledge Discovery and Data Mining (SIGKDD'08), Las Vegas (NV, USA)*, pages 731–739. ACM.

Using Healthcare Planning Features to Drive Scientific Workflows on the Web

Bruno S. C. M. Vilar, André Santanchè and Claudia Bauzer Medeiros

IC - UNICAMP, 13083-852, Campinas, SP, Brazil
{bvilar, santanche, cmbm}@ic.unicamp.br

Keywords: Scientific Workflows, Context-Adaptation, Task-Network Model, Healthcare.

Abstract: Automated healthcare planning (care-flow) systems are usually designed to afford the dynamicity of health environments, in which changes occur constantly as a patient's treatment progresses. This dynamic adaptation mechanism is based on blocks of activities, triggered and combined according to contextual data, producing a plan, which emerges from the interaction between these blocks and the context. However, tools that implement care-flow systems are still incipient, missing support for features like extensibility, collaboration and traceability of procedures. On the other hand, these features can be found in workflow systems that are widely used in a variety of environments (in business and scientific domains), with consolidated standards and technologies. However, workflow systems are not well suited to address the dynamicity of healthcare environments. In this paper we argue that care-flow and workflow systems have complementary characteristics and we present a software architecture that incorporates the emergent and context-driven approach of care-flow systems into workflow systems. We present a prototypical implementation validating the key concepts of our proposal, which uses an ontology representation of workflows combined with an ontology and SWRL rules.

1 INTRODUCTION

The growth in the number of patients of a hospital brings the challenge of managing appointments, admissions and surgical interventions. There is the need to increase the efficiency and flexibility to manage associated data. The use of computational resources to address this challenge implies on more technical requirements. The possibility of using data available on the Web has added a new dimension to this problem, with additional heterogeneity factors.

Some researchers (Unertl et al., 2009), in the context of chronic disease care, identified requirements that should be fulfilled by systems to be applied to the health domain, among which we single out: (i) Support for the shared needs and behaviors in care; (ii) Allow customization for disease-specific needs and to support the needs of different types of users; (iii) Explore new approaches for information input into the EHR (Electronic Health Record) as well as transfer, efficiently, data from medical devices into them.

Care-flow is at the core of healthcare management, involving directly or indirectly the requirements presented by (Unertl et al., 2009). As a consequence, one natural approach to start to deal with the problem has been to use Computer-Interpretable Guide-

lines (CIGs). Informally, a CIG is a specification in some kind of computer interpretable language that defines the flow of steps to be taken in each situation met by a health professional. Those guidelines manage the care-flow, customize needs and details for patient treatment and deal with data gathered, clinical guidelines, while preserves the rationale of healthcare professionals. Though CIGs are suitable for guiding the care-flow, from the perspective of the planning of the paths chosen, the tools that implement a CIG approach are still incipient, missing support to extensibility, reuse and share of content, collaboration among professionals and traceability.

Even though it is possible to define guidelines for "blocks of actions", the whole process involved in the healthcare of a given patient is driven by the context, which is captured during the process itself. In a typical scenario, data from given care-flow step will define the next steps to be followed. The flow of actions *emerge* from the interaction between available blocks and the context. Therefore, one of the most successful approaches for CIG is the Task-Network Model (TNM) (Peleg et al., 2003), which mimics this context-driven evolution. The process starts by a seed block of actions, which is unfolded according to context data collected during its execution.

Scientific workflows, on the other hand, are concerned with the planning and flow of tasks, but associated to scientific experiments and general tasks. They have been developed for years and tested on areas such as Astronomy, Biology, Gravitational Physics and Earthquake Science (Bharathi et al., 2008). As a result, those tools have consolidated mechanisms to deal with large amounts of data, collaboration, extensibility of their resources, and association with external sources of data, tools and algorithms. In general, Scientific Workflow Management Systems (SWfMS) manage experiments' provenance, keeping detailed and meaningful records of data involved on experiments and processes that affect the data. Provenance is fundamental for scientific processes because it provides important documentation that is key to preserve the data, to determine data quality and authorship, and to reproduce as well as validate the results of such processes (Davidson and Freire, 2008).

Workflow execution has been adopted where resources are distributed on the Web, helping the coordination and monitoring of processes. They are being increasingly used to create complex Web applications by Web service composition or by providing a thin client, accessible through a browser, to conduct large scale processes and experiments (Wei Tan, 2013). In order to enhance their applicability, research efforts are concerned with providing workflow mechanisms with more flexibility, including in the healthcare scenario such as (Dang et al., 2008), or (Schick et al., 2012). Even though there are mechanisms that can be used to pre-define exceptions and alternative paths along workflows, they were not designed to dynamically evolve the workflow specification driven by the context, e.g., unfolding blocks of actions according to contextual data collected during a process execution.

Compared to CIG systems, workflow systems have been widely adopted and have consolidated standards and tools. In order to exploit the advantages of workflow systems in the health context, this work proposes to incorporate into workflows the dynamic and context-driven approach followed by care-flow systems. This is based on our experience (Vilar et al., 2013) that shows that the effort to do this is less complex than the process to adapt CIG systems so that they can acquire the features that are causing scientific workflows to become widely adopted. This paper analyses the features that confer high flexibility to CIGs – mainly those based on the TNM due to its emergent behavior – and presents our initial approach to bring them to a SWfMS.

2 CLINICAL PRACTICE GUIDELINES AND TASK-NETWORK MODELS

Clinical Practical Guidelines (CPGs) are written guidelines that describe the evidence-based procedures to be followed during diagnosis, treatment, and clinical decision making for a specific disease. Their textual format can be easily diffused, but not easily used in daily work (Panzarasa and Stefanelli, 2006), because of a multitude of forms and specialized vocabulary. Moreover, they depend heavily on the expertise of health professionals.

An approach to solve these problems is the dissemination of guidelines' content in machine-interpretable representations, which are more suitable for use in individual clinical decision support (Panzarasa and Stefanelli, 2006). This approach has led to the creation of Computer-Interpretable Guidelines (CIGs), which implement guidelines in active computer-based decision support systems. CIGs adopt models to represent the content to support decisions. Some examples of such models are Task-Network Models (TNMs), Medical Logic Modules (MLMs) and Augmented Decision Tables (ADTs).

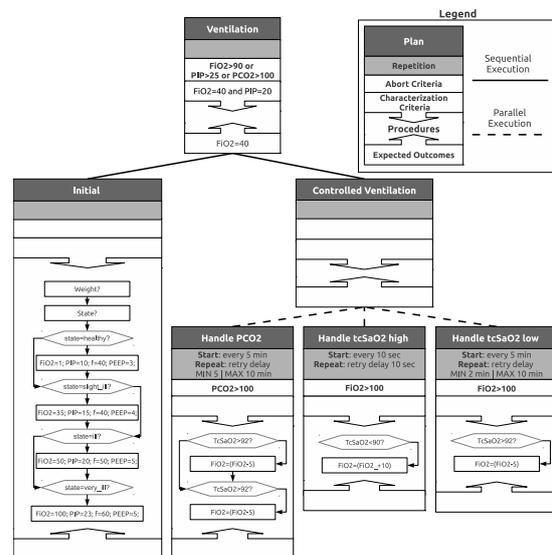


Figure 1: Example of CIG structure for a *Controlled Ventilation plan*.

(Gooch and Roudsari, 2011) identified 8 knowledge models implemented, related to clinical decisions, concluding that TNM was the most commonly adopted. The authors characterize TNM in two ways: general and formal. General TNMs are “flowcharts or process maps without formal semantics”. Formal TNMs are “guideline-based clinical tasks – ac-

tions, decisions, queries – that unfold over time, with a formal syntax and semantics”. According to (Pegleg et al., 2003), TNMs succeed over alternative approaches, such as MLMs and ADTs, because they do not provide full support for conceptualizing a multi-step guideline that unfolds over time. Considering those aspects, in this work we focus on TNMs.

Figure 1 shows a usual structure of a TNM. As the ‘Legend’ (top right) shows, the basic component is a plan and it represents a procedure to be applied to some case. In each box, a rectangle with a label is the name of the plan. The top plan – labeled *Ventilation*– is the starting (seed) plan. The following plans (below) are sub-plans that can be triggered by the upper plan according to rules. To decide whether a plan should be applied or not, there are *Characterization Criteria* that define conditions to be analyzed in order to match the patient case to the diagnosis. Another aspect that can be used to determine if the plan is suitable to the case is the *Expected Outcomes*. The flow of actions and conditions appear within the *Procedures* box. Whenever a plan is recognized as a way to achieve the same result desired by the healthcare professional, the plan will be triggered.

A triggered plan will follow the *Procedures* that are recommended to be applied to the patient. The repetition of those internal procedures is guided according to the *Repetition* specification associated to the plan. Also, it is possible to associate *Abort Criteria* to interrupt the plan when a specific situation is achieved. A plan can trigger potential sub-plans, allowing to modularize and reuse plans that already encapsulate needed practices. The fact that the results can be unexpected, and consequently the sub-plans will be selected during the execution, produces an *emergent behavior* in the flow of activities. Several potential flows, constrained by rules, will dynamically shape one flow on-the-fly during the execution, interacting with contextual values. This work captures this rule-based and context-driven emergent behavior and adapts it to workflows.

To explain the characteristics of TNMs we use the same scenario that Aigner and Miksch (Aigner and Miksch, 2006) adopt to validate the CareVis platform: Infants Respiratory Distress Syndrome (I-RDS). As Miksch (Miksch et al., 1998) explains, “after I-RDS is diagnosed, a plan dealing with limited monitoring possibilities is activated, called *initial-phase*. Then follows, depending on the severity of the disease, three different kinds of plans, *controlled-ventilation*, *permissive-hypercapnia*, or *crisis-management*. Only one plan at a time can be activated, however the order of execution and the activation frequency of the three different plans are depending on the severity of

the disease”.

The scenario is shown on Figure 1. The plan execution order is read as top-down, left to right. Dashed lines from a plan represent alternative sub-plans that are used according to the conditions. To represent the internal *Procedures*, we use the same representation as CareVis: flow-charts. An internal procedure is described as a flow of CareVis ‘single-steps’. Each single-step is either a variable assignment, a if-then-else construct (hexagon), an ask element (rectangle).

The *Ventilation* plan is activated when the characterization criteria $FiO_2 > 40$ and $PIP = 20$ is satisfied. The plan and all sub-plans are aborted if the condition $FiO_2 > 90$ or $PIP > 25$ or $PCO_2 > 100$ is achieved. As *Ventilation* plan does not have internal procedures, the *Initial* plan is executed, following the sequential order specified. The *Initial* plan just has a set of internal procedures, such as ask *weight* and *state* of the patient and define variable values according to the state value (*healthy*, *slightly ill*, *ill* and *very ill*).

3 WORKFLOWS

A workflow is defined as the movement of tasks through a work process describing how tasks are structured, who performs them, the resources needed and their relative order (Dallien et al., 2008). On a computational context, a workflow specification can be seen as an abstraction that allows the structured composition of programs as a sequence of activities aiming a desired result (Ogasawara et al., 2009).

Business and science are the two main driving forces behind the development of Workflow Management Systems (WfMSs). According to Sonntag et al. (Sonntag et al., 2010), business workflows are focused on the control of the flow, adopt agreed-upon communication standards, in order to facilitate interoperation between different software systems and companies, and commonly are concerned about fault handling, transactions, or quality of service features. Scientific workflows, on the other hand, are focused on data transformation, commonly may involve computation-intensive tasks, and are concerned with the specification of explicit data flow, the exact reproducibility of workflows, or processing of data streams.

The Workflow Management Coalition (WfMC) is one important participant of those driving forces behind WfMSs development. Because of its involvement on the earlier stage of the creation of workflows, it defines the workflow components under the business perspective. In this work we use the WfMC definitions, relaxing them to a broader

perspective that also covers scientific workflows (WfMC, 1999). Here we differentiate between a workflow specification and its instantiation (an actual execution). Moreover, we use the term ‘task’ as a synonym for ‘activity’ (where an activity is “a description of a piece of work that forms one logical step of a process” (WfMC, 1999). We likewise distinguish *Process definition* from *Process/Activity Instance*.

In order to help compare workflow and TNM approaches, we re-engineered the *Controlled Ventilation* plan of Section 2 using part of the graphical notation of flow-charts used on TNMs, but following a representation similar to that adopted by the Taverna (Hull et al., 2006) workflow system – see Figure 2.

The ‘legend’ (top right) shows the basic notion of structure of a process. The top rectangle with a text is the name of the process. The second one is the repetition criteria, which allows to define criteria to repeat the process and an interval for each repetition. Like TNMs, workflows may contain internal procedures, specified program code or instructions, that allows to perform activities. For instance, processes ‘Handle PCO2’ contains repetition criteria and embeds code that is repeated according to the criteria.

An important distinction between TNMs and workflows is the access to the variable values. While on TNM each value may act as a global variable, which can be accessed from anywhere in a plan, on workflows the values should be explicitly passed for each process through a port. Input ports receive the values of a process, while output ports contains the result of a process. Of course, processes can read and write from a common storage, but this approach reduces the generality of processes.

To exemplify the use of a workflow, consider again the *Controlled Ventilation plan*, now executed as a workflow depicted in Figure 2. To determine whether the *Ventilation process* should be executed, the *Ventilation Activation Condition* process is introduced. If the condition $FiO_2=40$ and $PIP=20$ is satisfied, the result is passed to the *Pass Output port*, which indicates that the *Ventilation process* should be performed. If the condition is not satisfied, the result is passed to *Fail Output port*, finishing the execution of the workflow. Ports thus enable conditioning the path according to the obtained result.

Considering that the *Ventilation Activation Condition process* generated a *Pass Output* value, the *Ventilation process* receives an input value and passes it to its internal processes. As well as on Figure 1, *Initial* and *Controlled Ventilation* are sequentially executed. This execution is explicitly defined by the fact that the

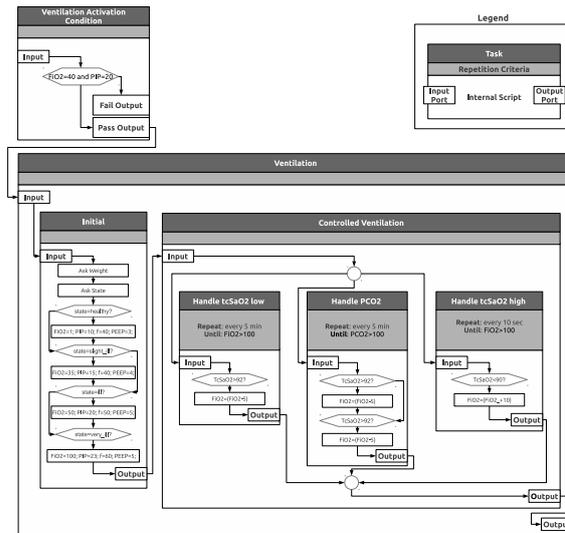


Figure 2: *Controlled Ventilation plan* re-engineered by us as a workflow structure.

Initial Output port is connected to the *Controlled Ventilation Input port*. While the *Initial process* uses its internal script to obtain the values of *weight* and *state*, *Controlled Ventilation* has sub-processes that are performed on any order. In fact, differently from TNMs, those sub-processes can be executed at the same time. The split-join, represented by a circle before and after the processes, indicates that any process (among *Handle tcSaO2 low*, *Handle PCO2* and *Handle tcSaO2 high*) can be executed. The split-join allows to execute different processes and then combines the results so that can be passed to another process or port.

If we proceed mapping a TNM specification to a workflow, the alternatives in each stage will grow exponentially. This effect shows the limits of trying to directly map a TNM specification in a classic workflow specification. Our proposal addresses this problem. It blends the TNM rule-based context-driven mechanism in a workflow system, providing an equivalent TNM ability of adapting itself according to the context.

4 DATA-DRIVEN VS PROCESS-DRIVEN APPROACHES

In order to summarize some of the main distinctions between the Workflow and TNM approaches is: scientific workflows are process driven and TNMs are data driven. Thus, bringing TNM characteristics into scientific workflows will make the latter both data and process driven.

TNMs are built to reduce the complexity to deal with large amounts of content that must be interpreted and to guide the use of procedures that are recommended to achieve some state. Because a patient may change his/her condition unexpectedly, TNMs allow to activate or deactivate content (guides and procedures) to treat the current state of the patient. All guides and procedures are pre-specified according to the recommendations of medical councils and well established procedures.

Scientific workflows are focused on processing high amounts of data. Usually there are two common situations: (i) a scientist wants to represent and automate a well known/consolidated experiment and execute it with different data (parameters) with none or few changes during the execution; (ii) the scientist wants to create an experiment to test a new hypothesis, so (s)he starts to build a workflow including and changing processes and parameters after different executions to analyze the results.

The difference between both approaches can be observable on the basic component of each one. TNMs have plans that associate content, procedures and other plans. Also, there are criteria that must be satisfied to activate the plan; such criteria allow selecting a plan over all other possible situations that may occur. Scientific workflows have processes that can be associated to others to create the flow of data and transform/process it. The data that is passed from one process to another can lead to different paths, but commonly the possible number of paths and changes are not as high as on TNMs. When a new situation occurs and there is a need to incorporate a new process, the scientist changes the workflow – e.g., choosing from a repository of processes, including an external tool or creating a new process.

Another factor is that, to deal with all resources involved, it is necessary to provide tools to visualize and filter information, as well as be aware of the origin and the involvement of resource with respect to the data. A key issue is to maintain provenance information on data and procedures – e.g., what kind of data was used where, how and by whom. Here, workflow systems already provide some sort of traceability mechanism.

In this paper we focus on the main features required to achieve a data-driven workflow: changing workflow tasks to adapt them to a situation. Our approach is presented next.

5 TOWARDS MAKING SCIENTIFIC WORKFLOWS DATA-DRIVEN

As presented in previous sections, a key factor to produce workflows able to adapt to the high dynamic health environments is the rule-based TNM approach, which triggers modules according to context values, producing an emergent behavior. Different from TNM specifications, which start from a seed and expands the flow, a workflow has a predefined starting flow. Therefore, we translated the mechanism of *unfolding* new activities on-the-fly according rules (TNM) to a mechanism of *adapting* the existing workflow on-the-fly according to rules (our approach).

In order to fully support dynamic workflow changes according to context, and give more flexibility to workflow execution, we decided to adopt reasoning capabilities and combine them with domain semantics. The solution found was to (a) use ontologies as the basis for workflow and concept representation, and (b) create adaptation rules to represent experts' knowledge. An *adaptation rule* characterizes a situation (cause) and defines the change (consequence) that should be performed to workflow so that it can be adapted to the context.

Given that there exist distinct workflow representations and formats (according to the WfMS adopted), the overall dynamic adaptation cycle can be described as follows. The workflow execution on the Web is monitored at each activity. Every new workflow state is mapped to an ontology instance, to which reasoning is applied, resulting in a new ontology instance (containing recommended modifications to that state). This new instance is transformed back to the workflow representation of the WfMS adopted, which is shown back to the user on the Web interface, to be validated or modified.

The architecture of our work is presented in Figure 3, where CRec (Context Record) is the linearized representation of an ontology, representing a workflow state. The main components are the following. A **Scientific Workflow Management System (SWfMS)** used to create and execute workflows, through a Web Interface, extended to incorporate the adaptation mechanism. A **Context-Aware Extension**, responsible for monitoring workflow execution and identifying the events that occur during that execution. A **Semantic Mapper** that is responsible for translating a workflow to a CRec and vice-versa, allowing the *Context-Aware Extension* to update the original workflow on SWfMS. An **Ontology** that specifies the main classes used to represent workflow activities and their parameters. Also, there are additional classes used to

complement activity information (such as activation status) and associated content (for instance, patient data). A CRec (resp. CRec') is a **Context Record** that is an ontology instance that represents the current state of a workflow and all information regarding the patient under care. A set of **Adaptation Rules (AR)**, written in SWRL, that will be used with an Ontology and a CRec to identify the changes recommended to the current situation. A **Context Adaptation Engine (CAEng)** that uses an Ontology Reasoner to process the AR combined to an Ontology and a CRec. The reasoning updates CRec to guide the adaptations that should be made on workflow.

The architecture enables dynamic adaptation of workflows to a context as follows. A SWfMS is accessed through a Web Interface (Figure 3, interaction 1). A workflow is assigned to be executed on the SWfMS (on interaction 2) and the SWfMS loads it (interaction 3) and updates the Web Interface (interaction 4). A user requests the SWfMS to start the executing workflow (interactions 5 and 6). When an activity is performed, the *Context-Aware Extension* captures the workflow state (interaction 7) and passes it to the *Semantic Mapper* (interaction 8). Next, the *Semantic Mapper* uses the workflow state to create an ontology instance that represents it (interaction 9) – CRec. The CAEng combines AR, Ontology, and CRec to identify the adaptations recommended to the workflow and apply them to CRec, creating a new version of it (CRec'). The *Semantic Mapper* receives CRec' (interaction 10) and maps its data to the workflow representation, passing it back to the *Context-Aware Extension* (interaction 11). The workflow on the SWfMS is updated by the *Context-Aware Extension* according to the new workflow (interaction 12). The SWfMS updates the Web Interface (interaction 13). Finally, the user sees the adapted version of the workflow (interaction 14).

Our architecture may be used on different scenarios and on different SWfMS, but its components created must be changed according to the scenario (e.g., rules and ontology customization) and platform adopted (e.g., Semantic Mapper).

6 INSTANTIATION ON THE WEB

We applied the principles behind our architecture to the context of nursing care to attest the flexibility of execution of the workflows. The tasks used to construct the ontology and the rules were based on the PROCEnf system (Peres et al., 2009), developed and adopted by the hospital of the University São Paulo. PROCEnf is also in process of adoption at the our

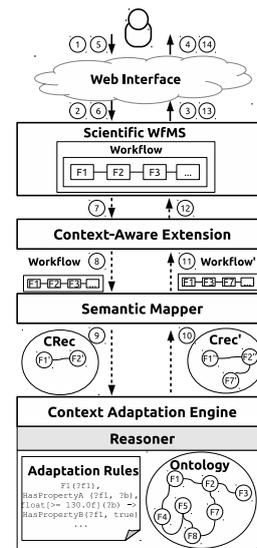


Figure 3: Architecture for context-based workflow adaptation.

University Hospital¹. Like most such systems, PROCEnf is heavily centered on offering health professionals sets of forms to fill.

We chose PROCEnf among other reasons, because we can reuse its components. Moreover, given our need to validate our implementation with actual users, this choice offers to the nursing staff of our University Hospital a set of forms and vocabulary they are familiar with.

Based on PROCEnf and on the work described in (Doenges and Moorhouse, 2008), we described the flow of a patient's admission and monitoring process in a hospital. In a general way, the process includes an iterative step which is started by an anamnesis interrogation, which is an assessment phase to evaluate the patients' conditions. Next, there is the analysis of recorded data to diagnose the problem and to identify expected outcomes (prognosis). Based on those expected outcomes, health interventions are planned and applied. To identify whether the outcomes were achieved or not, the intervention results are analyzed and the anamnesis records are updated. If the treatment achieves the expected outcomes, the patient can be released. Otherwise, a new iteration occurs.

To model PROCEnf forms as workflow activities, we created an ontology. In our current stage of research, we have about 30 activity classes involved with the diagnosis phase. Those classes are directly related to the PROCEnf system. As can be seen on

¹The hospital complex of the University of Campinas alone receives about 500,000 appointments, with over 43,000 admissions and 34,000 surgical interventions per year.

classes ‘CardiacFunction’ and ‘VitalSigns’, there are different attributes that can be filled to draw a diagnosis. Depending on how an attribute is filled, distinct decisions have to be taken – i.e., the workflow has to be changed dynamically.

Another issue we faced was the integration of workflow structures, rules, and reasoning capabilities. As explained before, this was solved by treating all information within a single ontology-based reasoning framework. The *Semantic Mapper* transforms the workflow structures (together with a current workflow state) into an ontology instance, that is enhanced with the current patient state. This is then treated by the reasoner as any other ontology, using adaptation rules that encode domain and user knowledge. Such adaptations are performed based on the adaptation rules we created in SWRL. Those rules can be classified into four types: I) **Propagate field (parameter) value**: repeat the value of a field to related fields, avoiding the need to fill the value in other forms. II) **Infer field (parameter) value**: the value of a field according to the value of other fields. III) **Change form (activity) position**: the order in forms which are filled can be changed, increasing or decreasing filling priority, making related forms closer and unrelated or unlikely forms more distant in the filling order. IV) **Include/Remove form (activity)**: a form can be removed when a field makes it incompatible with the purpose of the form (e.g., pregnancy issues related to men). Inclusion occurs whenever a situation makes a previously removed form viable.

Table 1 presents a subset of the rules, highlighting the types and some of their potential. The rules are specified in SWRL and are specified in cause/consequence form. The cause (antecedent) can be defined according to the value of fields and forms, by the composition of different predicates. A consequence (consequent) is the result of checking the antecedent, e.g., changing of a property value, including or removing values. For instance, rule *R2* says that when the *Systolic Blood Pressure* value of *Vital Signs* is large than 130, then the field *Has Systolic Blood Pressure Out of Expected Range* should be true.

Let us exemplify how adaptation rules are used in the reasoning process. Consider the following scenario: a nurse should fill a set of forms about different patient conditions, including vital signs, self-care, and comfort. At the beginning of the process, the professional fills a form about *Vital Signs*. If the value of *systolic blood pressure* is filled with value large than 130, rule *R2* (Table 1) has its antecedent condition satisfied, so the inference engine applies the consequent which characterizes the patient as having a *systolic pressure out of expected range*. As a re-

sult, two other rules are to be executed, leading to the new adaptations: *R1*, to propagate the value of *systolic blood pressure* to the *Cardiac Function* task and *R4* to increase the priority of the *Cardiac Function* task, which will put this form closer to the *Vital Signs* form. This order change highlights the need to provide additional information about Cardiac Function details and to avoid the loss of focus and information regarding this problem.

7 CONCLUSIONS

In this paper, we presented a proposal to adapt scientific workflows to the context of healthcare management, and its Web implementation. Our modifications to an SWfMS place such systems closer to the domain of healthcare, thanks to the inspiration from CIGs and to the fact that such systems, thanks to the data-driven flow, are suitable to dynamic scenarios.

We designed the model in which an SWfMS is extended by a layer which uses ontologies combined to rules as a means to make scientific workflows more data-driven than just process-driven. This in turn, brings more flexibility to the execution of the tasks as well as allows to choose whether a task should be executed according to more sophisticated conditions.

Future steps of this research will cope with TNM features, presented in Section 4, working on the better graphical integration and adaptation of the work with the SWfMS, allowing to: (i) Integrate the ontology content to the workflow tasks, allowing users to navigate through the concepts of ontologies and use those concepts also as a way to filter tasks and conveniently find resources on SWfMS; (ii) Provide support to the hierarchical organization of workflow tasks, giving more dynamic flow to task execution as well as making it more similar to the TNM approach.

ACKNOWLEDGMENTS

Work partially financed by FAPESP (grant 2011/17196-0), FAPESP/Cepid in Computational Engineering and Sciences (2013/08293-7), the Microsoft Research FAPESP Virtual Institute (NavScales project), CNPq (MuZOO Project), FAPESP-PRONEX(eScience project), INCT in Web Science, and individual grants from CNPq. We also thank healthcare providers from University of Campinas and University of São Paulo, who provided us support in this research.

Table 1: Adaptation Rules.

| Rule | Antecedent | Consequent | Type |
|------|--|--|-------------------|
| R1 | CardiacFunction(?cf), VitalSigns(?vs), systolicBloodPressure(?vs, ?x) | systolicBloodPressure(?cf, ?x) | Propagate Value |
| R2 | VitalSigns(?vs), systolicBloodPressure(?vs, ?x), float[>= 130.0f](?x) | hasSystolicPressureOutOfExpectedRange(?vs, true) | Infer Value |
| R3 | VitalSigns(?vs), isActive(?vs, true), hasSystolicPressureOutOfExpectedRange(?vs, false), CardiacFunction(?cf) | position(?cf, 99) | Decrease Priority |
| R4 | VitalSigns(?vs), hasSystolicPressureOutOfExpectedRange(?vs, true), isActive(?vs, true), CardiacFunction(?cf), position(?vs, ?po) | position(?cf, ?po) | Increase Priority |
| R5 | ValuesAndBeliefs(?va), Evaluation(?e), hasPatient(?e, ?p), isActive(?e, true), hasAge(?p, ?age), integer[<= 12](?age) | isRemoved(?va, true) | Remove |
| R6 | ValuesAndBeliefs(?va), Evaluation(?e), hasPatient(?e, ?p), isActive(?e, true), hasAge(?p, ?age), integer[> 12](?age), isRemoved(?va, true) | isRemoved(?va, false) | Include |

REFERENCES

- Aigner, W. and Miksch, S. (2006). Carevis: Integrated visualization of computerized protocols and temporal patient data. *Artificial Intelligence in Medicine*, 37(3):203–218.
- Bharathi, S., Chervenak, A., Deelman, E., Mehta, G., Su, M.-H., and Vahi, K. (2008). Characterization of scientific workflows. *2008 Third Workshop on Workflows in Support of Large-Scale Science*, pages 1–10.
- Dallien, J., MacCaull, W., and Tien, A. (2008). Initial work in the design and development of verifiable workflow management systems and some applications to health care. In *Proceedings of the 2008 5th International Workshop on Model-based Methodologies for Pervasive and Embedded Software*, MOMPES '08, pages 78–91, Washington, DC, USA. IEEE Computer Society.
- Dang, J., Hedayati, A., Hampel, K., and Toklu, C. (2008). An ontological knowledge framework for adaptive medical workflow. *Journal of biomedical informatics*, 41(5):829–36.
- Davidson, S. and Freire, J. (2008). Provenance and scientific workflows: challenges and opportunities. In *SIGMOD Conf.*, pages 1345–1350. Citeseer.
- Doenges, M. and Moorhouse, M. (2008). *Application of nursing process and nursing diagnosis: an interactive text for diagnostic reasoning*. G - Reference, Information and Interdisciplinary Subjects Series. F.A. Davis.
- Gooch, P. and Roudsari, A. (2011). Computerization of workflows, guidelines, and care pathways: a review of implementation challenges for process-oriented health information systems. *Journal of the American Medical Informatics Association : JAMIA*, 18(6):738–48.
- Hull, D., Wolstencroft, K., Stevens, R., Goble, C., Pocock, M. R., Li, P., and Oinn, T. (2006). Taverna: a tool for building and running workflows of services. *Nucleic acids research*, 34(-):W729–32.
- Miksch, S., Kosara, R., Shahar, Y., and Johnson, P. D. (1998). AsbruView: Visualization of Time-Oriented, Skeletal Plans. pages 11–18.
- Ogasawara, E., Paulino, C., Murta, L., Werner, C., and Matoso, M. (2009). Experiment line: Software reuse in scientific workflows. In Winslett, M., editor, *Scientific and Statistical Database Management*, volume 5566 of *Lecture Notes in Computer Science*, pages 264–272. Springer Berlin Heidelberg.
- Panzarasa, S. and Stefanelli, M. (2006). Workflow management systems for guideline implementation. *Neurological sciences : official journal of the Italian Neurological Society and of the Italian Society of Clinical Neurophysiology*, 27 Suppl 3:S245–9.
- Peleg, M., Tu, S., Bury, J., Ciccarese, P., Fox, J., Greenes, R. A., Miksch, S., Quaglini, S., Seyfang, A., Shortliffe, E. H., Stefanelli, M., and et al. (2003). Comparing computer-interprettable guideline models: A case-study approach. *JAMIA*, 10:2003.
- Peres, H. H. C., Cruz, D. D. A. L. M. D., Lima, A. F. C., Gaidzinski, R. R., Ortiz, D. C. F., Trindade, M. M. E., Tsukamoto, R., and Conceição, N. B. (2009). Desenvolvimento de Sistema Eletrônico de Documentação Clínica de Enfermagem estruturado em diagnósticos, resultados e intervenções. *Revista da Escola de Enfermagem da USP*, 43(spe2):1149–1155.
- Schick, S., Meyer, H., Bandt, M., and Heuer, A. (2012). Enabling yawl to handle dynamic operating room management. In Daniel, F., Barkaoui, K., and Dustdar, S., editors, *Business Process Management Workshops (2)*, volume 100 of *Lecture Notes in Business Information Processing*, pages 249–260. Springer.
- Sonntag, M., Karastoyanova, D., and Deelman, E. (2010). Bridging the gap between business and scientific workflows: Humans in the loop of scientific workflows. In *e-Science (e-Science), 2010 IEEE Sixth International Conference on*, pages 206–213.
- Unertl, K. M., Weinger, M. B., Johnson, K. B., and Lorenzi, N. M. (2009). Describing and modeling workflow and information flow in chronic disease care. *Journal of the American Medical Informatics Association : JAMIA*, 16(6):826–36.
- Vilar, B. S. C. M., Medeiros, C. B., and Santanchè, A. (2013). Towards adapting scientific workflow systems to healthcare planning. In *HEALTHINF - International Conference on Health Informatics*.
- Wei Tan, M. Z. (2013). *Business and Scientific Workflows: A Web Service-Oriented Approach*. Wiley.
- WfMC (1999). Terminology and Glossary Document Number WfMC-TC-1011 - Issue 3.0. Technical report, Workflow Management Coalition.

Linked Data Strategy to Achieve Interoperability in Higher Education

Guillermo García Juanes, Alioth Rodríguez Barrios, José Luis Roda García, Laura Gutiérrez Medina, Rita Díaz Adán and Pedro González Yanes

School of Computer Science, University of La Laguna, San Cristóbal de La Laguna, Spain
ggjuanes@gmail.com, alioth.riguez@gmail.com, jlroda@ull.edu.es, {lgutmed, ritadiazadan}@gmail.com, pgonyan@ull.edu.es

Keywords: Linked Data, Open Data, Interoperability, High Education.

Abstract: An important challenge in centres of higher education is the use of Linked Data strategy to connect currently existing multiple information systems. These information systems are usually independent from one another, and the ability to obtain information by connecting different sources of data involves, in most cases, unacceptable costs and effort. In this work, we have developed a platform based on Linked Data that permits the interoperability of different sources of data, both internal as well as external. This interoperability is achieved by 1) the use of higher education ontologies, and 2) the use of a process that begins with the analysis of the data sources to be connected, followed by mapping of the closest ontologies, and ends with the generation and publication of data in valid formats for Linked Data. The final product permits stakeholders inside and outside the university to be able to make queries of two or more datasets in different information systems at the same time.

1 INTRODUCTION

The term Open Data concerns offering to society the data collected by public institutions, which, when handled by third-parties, can be of great value for the development of applications, reports, etc. The principal objective of the Open Data strategy is to offer transparency, participation and collaboration in the publishing of information, in standard formats, open and interoperable, facilitating its access and permitting its re-use (Office, 2012). This is nothing more than data that belong to public administration being used, by individuals or companies, with or without commercial ends, provided that use does not constitute public administration activity. There are many institutions that currently publish open data in different formats (Fundación CTIC, 2013) (Bauer and Kaltenböck, 2012).

The objective of Linked Data is to give meaning to connections that are found in different datasets so that machines can obtain more relevant information making use of techniques from the Semantic Web (Berners-Lee, 2006).

The relationship between Open Data and Linked Data was proposed by Tim Berners-Lee who

suggested a way of measuring the degree of quality of data published from an Open Data portal, where, if those data were to be published using Linked Data principles, the highest degree that a portal may achieve would be obtained (Berners-Lee, 2006).

The data interoperability is achieved through the following of the Linked Data principles. These vocabularies must cover a wide range of concepts related to the university's system. From the institution, the departments, the teachers, including the courses, programmes and study material, credits, theory classes, practical classes and laboratory classes, etc., a vocabulary must cover all these aspects to be able to be linked and make full use of the Linked Data strategy.

In this work we present a prototype that was developed at the University of La Laguna (ULL), where various different groups played a part: the Planning and Analysis Office, the ULL Information Technology Service, and the Taro Research Group. The project concerns the demonstration of how Linked Open Data offers a range of benefits for the procurement of information that are not achieved through other conventional methods. The higher education system comprises multiple information systems, some of which are quite complex. This

paper is, therefore, concerned with the application of Linked Data strategy, methods and techniques to some of these university systems.

2 RELATED WORK

The development of a platform such as the one intended requires a prior state-of-the-art study, with particular emphasis in the area of ontologies that may be re-used with those that are going to model concepts within the scope of higher education.

In the search for ontologies closer to our problem, we made use of semantic search engines such as Watson¹ or Swoogle² apart from search engines for key words or known repositories like Linked Open Vocabularies (LOV)³.

There is a great set of ontologies related to the field of education, but we limited our search for ontologies to those that could be adapted as far as possible to our case. They were also evaluated for their quality based on whether they were structured, well documented and in current use by other organizations.

From this analysis, candidate ontologies were obtained for re-use in our system. The most relevant ones, related to universities, were Academic Institution Internal Structure Ontology (AIISO)⁴, Teaching Core Vocabulary Specification (TEACH)⁵ and The Bowlogna Ontology⁶. AIISO describes the organizational structure of the university very simply, showing the hierarchy and relationships between different agencies. In that regard, it provides classes for modelling: teaching staff, subjects, departments, centres, etc. The TEACH and Bowlogna ontologies describe the part more related to teaching, that is, they try to represent the relationship between a program, the subjects that comprise it, the teaching workload and responsibilities of teachers for each of the subjects. Bowlogna is a more special case, representing the organization of the university following the Bologna Plan currently being implemented by European universities (Demartini et al., 2013).

All things considered, a university is still an organization, aside from the academic element, which can be modelled with organizations

ontologies like W3C's The Organization Ontology⁷ or Buildings and Rooms Vocabulary⁸. Finally, we intend to define the situation for premises and staff, and CTIC's Vocabulary of Localizations⁹ or vCard Ontology¹⁰ can be used for that, providing a series of classes with which to model addresses, municipalities, provinces, etc. These ontologies are very important as in many cases they are standard and widely used, as their use is not limited to a specific field.

Another important action was to take examples of the strategies established by other universities as a step towards the publication of Linked Data. They usually follow a basic scheme, re-using as far as possible existing ontologies, and if these do not cover all of the necessary concepts, the ontology itself is created or extended from an existing one. This can be seen in the Open Data sections of the University of Oxford¹¹, the University of Southampton¹² and the Open University¹³, that usually have an Open Data portal from which you can consult information modelled in different forms (navigation, queries, etc.). These portals are usually based on RDF software, Virtuoso being the most popular, although there are other tools such as Pubby¹⁴, Fuseki¹⁵ or D2R¹⁶.

The general situation is that when choosing ontologies that adapt themselves more to universities, there is consensus for the use of several of those named above, but none ever achieves a complete representation of the information, which is why it is necessary to create another to be able to link it with the other ontologies.

To achieve the publication of data, the ontologies must be published in such a way that they are dereferenceable and well documented, permitting the rest of the world to re-use them, thus achieving interconnections and future inferences of information (W3C, 2008) (Heath and Bizer, 2011).

In our case, we do not enter into the creation of ontologies in depth, as that was not one of the principal objectives that we wished to demonstrate, but rather interoperability between systems that

¹ <http://watson.kmi.open.ac.uk/WatsonWUI/>

² <http://swoogle.umbc.edu/>

³ <http://lov.okfn.org/dataset/lov/>

⁴ <http://vocab.org/aiiso/schema>

⁵ <http://linkedsience.org/teach/ns/#>

⁶ <http://diuf.unifr.ch/main/xi/bowlogna>

⁷ <http://www.w3.org/TR/vocab-org/>

⁸ <http://vocab.deri.ie/rooms>

⁹ <http://purl.org/ctic/infraestructuras/localizacion>

¹⁰ <http://www.w3.org/TR/vcard-rdf/>

¹¹ <https://data.ox.ac.uk/>

¹² <http://data.southampton.ac.uk/>

¹³ <http://data.open.ac.uk/>

¹⁴ <http://wifo5-03.informatik.uni-mannheim.de/pubby/>

¹⁵ http://jena.apache.org/documentation/serving_data/

¹⁶ <http://d2rq.org/>

implement Linked Data as a procedure prior to publication of data, bringing it into a real context.

3 MOTIVATION AND GOALS

The University of La Laguna is a large-scale public higher education institution. It currently involves more than 26,000 people including students, teachers and administration staff, distributed between 25 centres, 60 departments and other areas. There are also close links with private entities (companies, foundations and institutes) and public institutions such as the local island administration Cabildo de Tenerife, City Councils and the Government of the Canary Islands.

The organizational structure is decentralized, and many functions are delegated to each of the departments, centres and services. This fact has a special relevance to this work, as we have had to study the functions of the principal organizational units in depth. Each unit works almost independently, each one takes responsibility for handling administrative processes that have been delegated to them, collecting and maintaining the information necessary to operate (accounting, academic administration, libraries, ITC centres, etc.).

With respect to this work, there are many information systems that offer support to the university as a whole. Financial Management and Academic Management are among the main ones that are found. The first system takes care of the management of the administration staff and the teaching staff; while the second is concerned with the enrolment processes and everything related to teaching. Both aggregate a large quantity of data and are more or less controlled forming a fairly homogenous architecture. They use the same group of software development technologies, the same database management system, and, most relevant, they are under the responsibility of the same IT department. Although there is a certain homogeneity between these systems, extracting information across both systems continues to be a complex and costly task.

Apart from larger, older systems, there are smaller, independent systems, of great value to the institution. These have appeared over time according to the needs of services or departments. Examples of these systems are: the research service, the directories of the institution's staff, quality control, diaries and events, etc. These are usually controlled by different areas and each one can have different

software.

It is at this point that this work begins to have meaning. There are many cases where management, statistical or other similar information is requested from other institutions within the university itself. Most of these systems work independently from one another, and when it is necessary to consult information from two or more sources, it is necessary to establish connections between the different systems. Due to the complexity and internal structure of each system, staff have to make a particularly strenuous effort to obtain the data.

The main motivation to work in this environment and to offer a practical solution to these problems based on Linked Data is, on the one hand, the existence of the real problem of access to different sources of data, and on the other, that the university's staff is quite able to accept new proposals and finally, the existence of a real and concrete problem with which to apply Linked Data methods and techniques (for the re-use of ontologies, generation of RDF links, publication of data and consumption of published data).

The key challenge is to offer the university a solution for offering information obtained from a variety of sources, without modifying the current systems, as many are legacy systems and are so assimilated within the institution that a change to them could cause chaos. This is the solution that we present below.

4 LINKED DATA PROCESS

The process of linking data from the different information systems has followed an iterative and incremental methodology (Suárez de Figueroa Baonza, 2010). This process has allowed us to obtain valuable results from the first iterations and refine them continuously. As a consequence, the Linked Data process is provided with the necessary flexibility to tackle the changes related to the requirements in any phase of the process.

The working methodology comprises six differentiated phases inspired by methodologies commonly used for the publication of Linked Data (Poveda-Villalón, 2012); (Corcho et al., 2013); (Fernández-López, 1997); (Atemezing et al., 2012) and adapted to the needs of the particular working environment. These phases are divided between: specification of data to be published, data modelling, generation of data in RDF, publication, linking and exploitation.

Given the peculiarities of linking data in our

university context, as opposed to that proposed for different methodologies for the publication of data (Poveda-Villalón, 2012), the publication phase is undertaken at an earlier stage to the linking of data.

4.1 Specification

The first task to be dealt with is the specification of the data to be published. ULL has a large amount of data related to administrative, academic or financial activities. With the aim of demonstrating the utility of linking data in the University under the paradigm of Linked Data, and in view of the difficulty of linking all the data in the institution owing to the large volume, two clearly defined organizational units were chosen: the organizational area and the academic area.

The data available in each of these areas were not related with one another, although they made reference to entities that could be easily linked.

The data of the University’s organizational area describe the hierarchy of the institution, the structure of the teaching staff and the distribution of the courses and the teaching areas. Figure 1 presents the organizational domain model:

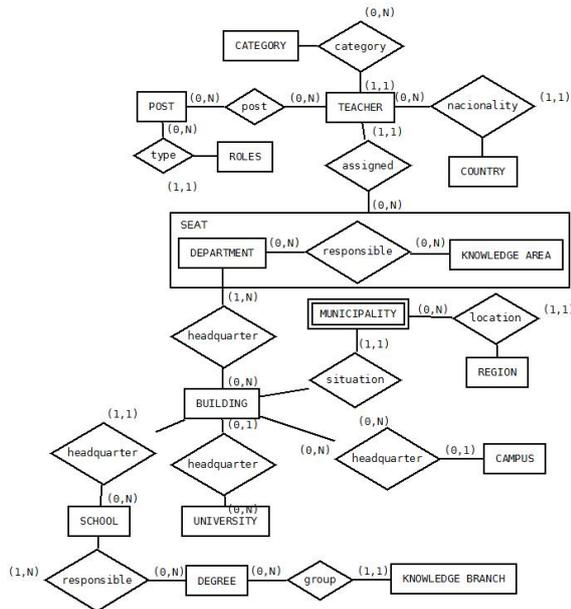


Figure 1: Organizational domain model.

The academic area contains information relating to the courses, which have been adapted to the Bologna Plan. The data from the academic area, Figure 2, contains information relating to the study plan of the courses, provided by the Spanish Government’s Ministry of Education, Culture and Sport, plus

information relating to the teaching staff teaching those courses.

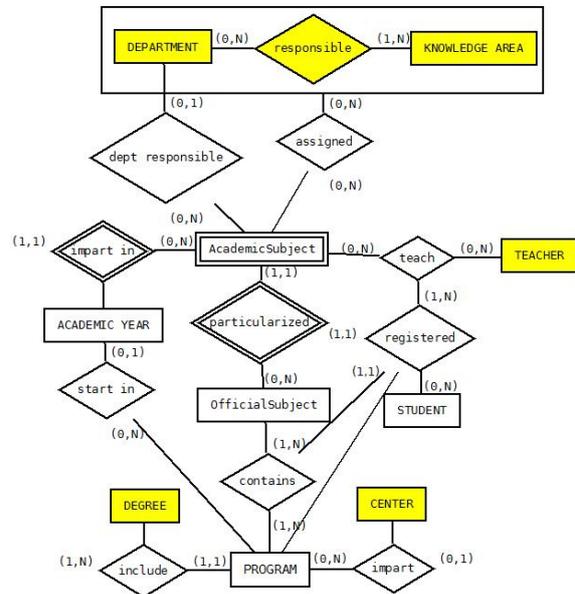


Figure 2: Academic domain model.

4.2 Modelling

Once the data to be linked had been determined, the next step was to analyse them in order to be able to begin modelling it with a set of ontologies.

Due to the peculiarities of the starting dataset, we opted for the development of an ontologies network, which had been re-utilized and created with ontologies with different characteristics. This ontologies network re-utilizes ontologies from the academic environment such as AIISO and TEACH, and more general and frequently used ontologies such as FOAF¹⁷, SKOS¹⁸ and others including LOC, ORG and ROOMS that model localization, and organization and premises, respectively. To complete the network, we created three new ontologies intended to represent the semantics which earlier ontologies could not cover. One of the ontologies was centred on academic information, and the other on organizational information, and the final one modelled general aspects of the institution. The development of the network of ontologies was carried out using NeOn Toolkit¹⁹, an open source tool based on the Eclipse platform. It provides a set of plug-ins that cover many of the needs arising during this cycle of ontology development, such as

¹⁷ <http://xmlns.com/foaf/spec/>

¹⁸ <http://www.w3.org/2008/05/skos>

¹⁹ <http://neon-toolkit.org/>

the generation of documentation, modularization, or the evaluation of ontologies (Zemmouchi-Ghomari & Ghomari, 2013). Figure 3 shows the network ontology obtained.

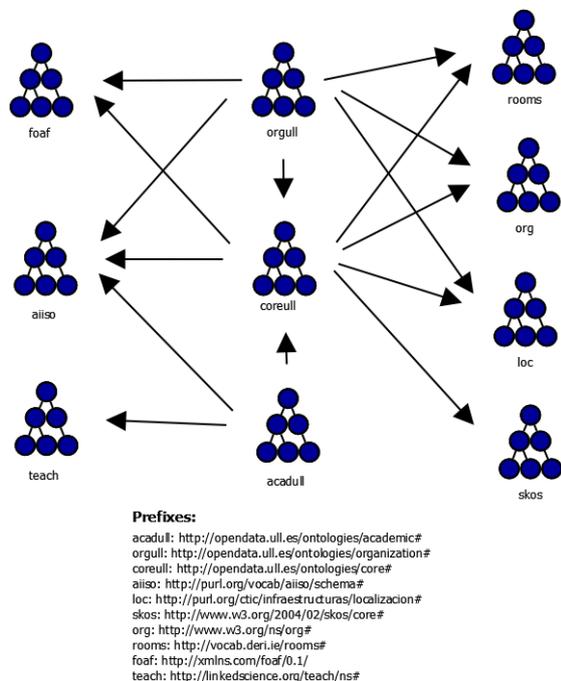


Figure 3: Modelled ontology network.

As part of the modelling phase, the anatomy of the URI is defined under the guidelines of the so called Cool URIs (W3C, 2008). As a result, the URIs of the resources follow the following pattern:

```
http://datos.uil.es/resource/{resource
type}/{resource}
```

Code 1: URI structure.

For example, the “IT Engineering” course is identified by the URI:

```
http://datos.uil.es/resource/programme/
IT_Engineering
```

Code 2: Example of an URI. IT Engineering identifier.

4.3 Generation

Once the ontology network has been defined, the next step is to generate the data in standard RDF format.

The D2RQ platform was used for that process, which allowed for data stored on relational databases to be consulted as if they were RDF graphs, making use of SPARQL language. Using a mapping

language, in which the transformations to be made were specified following the defined ontology, this tool makes it possible to obtain data in RDF format by directly consulting a relational database.

As a consequence of using a tool with these characteristics, the results of this phase did not consist of a set of data in RDF, but in a file with the definition of the correspondences or mappings between the different fields in the organizational and academic area databases, and the elements of the ontology network previously defined (see Figure 3).

An extract of the mapping file obtained during this phase can be seen below:

```
map:institutions a d2rq:ClassMap;
d2rq:dataStorage map:database;
d2rq:uriPattern
"institutions/@@UNIVERSIDAD.NOMBRE|urli
fy@";
d2rq:class ullorg:Universidad;
```

Code 3: Mapping an institution type resource in D2R.

4.4 Publication

The main objective of this phase is to obtain two access points for queries to the RDF format data of the organizational and academic areas.

Thanks to the implementation of a Pubby based interface, through the use of D2R, a front-end for SPARQL endpoints, access can be given to data stored in the institution’s relational databases through an HTML interface which displays them in RDF and at a SPARQL endpoint. This permits us to navigate through the information as well as to make specific queries.

4.5 Interlinking

Throughout this phase, we intended to achieve the objective of the availability of a five star datasets, following Tim Berners-Lee’s recognised classification, that is, a set of data linked with other sets of data (Heath and Bizer, 2012).

In our case, the linking process is divided into two phases, one of which establishes a data link internally, and another that links external data sources.

The internal linking, between our two access points, was achieved thanks to what was already known about how the URIs were going to be formed, as the structure is well defined and the identifiers of each of the resources in most cases are stipulated beforehand, and are common to the whole organization. Due to this, it is not necessary to use tools to discover candidates. Therefore is only

necessary that each system can be able to obtain the identifiers from the other system to create the URIs. In order to do this, the systems retrieve them from the other source using a SQL script. This process is only needed when new resource appears and not each time that a query is done.

If this information was not held, it would be necessary to make public the same resources in both domains (to duplicate the occurrences) and then to use linking tools, with owl:sameAs properties, to indicate that they are the same resource.

The external linking is possible when external data sources exist. For that, we undertook an analysis, and due to the scarcity of reliable sources, we limited ourselves to linking with dbpedia.org and linkeddata.es.

This link entails a process in which candidate links have to be found, and we intended to automate it as far as possible, using a Silk tool²⁰ and Link Specification Language (LSL) configurations. Once the tool was implemented, results that corresponded with the external resources that could be linked to our data were generated. These links were added to the database *LINKED* table in the following form:

```
http://opendata.ull.es/resource/{resourceid} owl:sameAs http://{externalresource}
```

Code 4: Structure of sameAs statement.

This table is like a repository or cache where all the links that are found are stored and managed, containing: the URI of the internal resource, the URI of the external resource and the property that links them. In this way we are able to make links not only using owl:sameAs but also other useful properties, as well as management fields for configuration issues. This table also allow us to determine if a link is valid, if has been manually blocked, the date that the link was discovered, etc.

Once the links are added, the information is republished automatically with the external links. This is thanks to the added dynamic property in the D2R map that takes care of reviewing the linked table and showing the properties that exist there:

```
d2rq:belongsToClassMap map:CLASSTOLINK;
d2rq:dynamicProperty
"@@linked.propiedad@";
d2rq:uriColumn "linked.objeto"; .
```

Code 5: Mapping sameAs statement in D2R.

²⁰ <http://wifo5-03.informatik.uni-mannheim.de/bizer/silk/>

4.6 Exploitation

When addressing this phase, we started from the two SPARQL access points available for the consultation of data from the University's organizational and academic areas.

The objective of this phase was the availability of a single point of access to the platform that would permit federated queries on the two groups of data. To achieve that objective, a Fuseki server was deployed, allowing serve data in RDF format using HTTP. This service offers data to users through a RESTful API and an enriched SPARQL endpoint, from which so called SPARQL++ queries may be made (Polleres et al., 2007). With this service, complex queries can be made using proprietary databases or external endpoints. Previously, these types of queries were only possible using scripts and by the acquisition of data from sources outside the organization.

Here is an example of a complex query making use of two data sources:

```
PREFIX coreull: < ... >
PREFIX orgull: < ... >
PREFIX acaull: < ... >
SELECT ?NameDept (COUNT(?planning) as
?numberPlanning) WHERE {
    SERVICE <http://orgull/sparql> {
        ?teacher orgull:miembro ?dept .
        ?dept coreull:nombre ?NameDept .
    }
    SERVICE <http://acadull/sparql> {
        ?planning acaull:tieneProfesor
?teacher .
    }
} GROUP BY ?NameDept
ORDER BY ?NameDept .
```

Code 6: Query example: "How many subjects does each department teach?"

5 FINAL PRODUCT

The project finally developed is presented in Figure 4. We reproduced the complete system on a development server simulating the real system used at ULL. As it can be seen in the figure, we have two independent information systems: Academic and Organizational Management. Both systems reside in different databases supported by MySQL. An architecture based on D2R was developed above each of the two current systems for the mapping of the relational database to RDF files. Each system physically resides in a different server and has,

therefore, a separate portal for each environment.

Each D2RQ server had an SPARQL endpoint added and a web interface (Pubby-type) where corresponding information could be obtained.

A supervised process to discover and update external links was added using the Silk tool. Moreover, Fuseki service was also added as a junction for the whole system, enabling SPARQL++ queries.

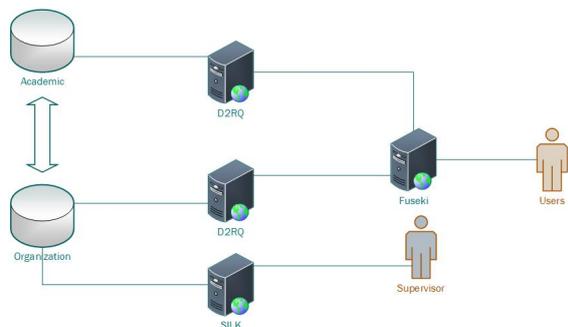


Figure 4: Architecture diagram.

Finally, it was possible to obtain an open data publication portal based on web semantics, meeting Linked Data requirements. Users will be able to use it to consult University information as well for making queries at several SPARQL points (so called SPARQL Endpoints) that may be linked to the published data. An example of this would be making a query such as “Distribution of students by province and their geo-localization to be shown on a map”, represented in SPARQL as:

```

PREFIX coreull:
<http://orgull/ontologies/core#>
PREFIX orgull:
<http://orgull/ontologies/organization#>
PREFIX acadull:
<http://orgull/ontologies/academic#>
PREFIX georss:
<http://www.georss.org/georss/>

SELECT ?Region (COUNT(?student) as
?numberStudents) ?geoloc
WHERE {
    SERVICE < http://acadull/sparql >
    {
        ?student a acadull:Alumno;
        coreull:provincia ?reg .
    }
    SERVICE < http://orgull/sparql > {
        ?reg coreull:nombre ?Region ;
        owl:sameAs ?linked .
    }
    SERVICE <
    http://dbpedia.org/sparql > {
        ?linked georss:point ?geoloc .
    }
} GROUP BY ?Region .
    
```

Code 7: Query with external data sources. Recount of students and their geolocalization.

The final platform enables users with simple SPARQL queries to make cross-queries of data from different sources that were previously inaccessible or of very complex interoperability. The end users, both inside and outside the University, have available a tool based on Linked Data to obtain the information they need. If case of need, it would be possible to add a user interface to make its use more user-friendly.

6 CONCLUSIONS

In this work, we have developed a genuine platform based on the Linked Data strategy in which universities may publish data and link with internal and external sources. The interlinking of the different university data sources will permit greater interoperability and can achieve new knowledge from this relationship.

To deploy our prototype in the university production servers, it is necessary to consider two important aspects: the integration costs of our platform with the university systems and also the university staff training in linked data. We firmly believe that both aspects can be addressed as we have the necessary technical expertise.

Whit this project, the institution has a tool to make complex queries, which hitherto existing systems could only handle with great effort. Moreover, the same tool could be offered to other local, island, and regional institutions in such a way that they could make queries directly, rather than through a service request to university personnel.

The ontologies most related to higher education have been revised, and improvements have been proposed to cover concepts not covered by existing ontologies. The network of ontologies used in this work covers ten ontologies, of which only three have been newly created.

Finally, as a summary of the work undertaken, we would stress that effective analysis of the initial data produces a lower number of errors in the re-use and definition of ontologies. We would also indicate that the process of creation of ontologies was based on the basic terms needed to demonstrate the viability of the project. We will continue to research several concepts that are beginning to be requested, in more depth, once the first version of the platform has been validated. With respect to the technological platform, a viability study will have to be undertaken to compare the effort required to incorporate triple store systems like Virtuoso, etc.

We should not forget that any new system for the organization would also implicate the commitment of human resources and materials for its development and future maintenance.

ACKNOWLEDGEMENTS

We would like to thanks to Andrés Palenzuela from the Planning and Analysis Office of University of La Laguna in the Canary Islands. His interest and support has made this Project viable.

REFERENCIAS

- Atemezing, G., Corcho, O., Garijo, D., Mora, J., Poveda-Villalón, M., Rozas, P., Villazón-Terrazas, B. (2012). Transforming meteorological data into linked data. En *Semantic Web*. IOS Press.
- Bauer, F., & Kaltenböck, M. (2012). *Linked Open Data: The Essentials A Quick Start Guide for Decision Makers*.
- Berners-Lee, T. (2006). *Linked Data*. Recuperado el 11 de 10 de 2013, de <http://www.w3.org/DesignIssues/LinkedData.html>
- Corcho, O., Fernández-Lopez, M., & Gómez-Perez, A. (2003). Methodologies, tools and languages for building. Where is their meeting point? En *Data & Knowledge Engineering* (págs. 41–64). Elsevier.
- Demartini, G., Enchev, I., Gapany, J., & Cudré-Mauroux, P. (2013). The Bowlogna Ontology: Fostering Open Curricula and Agile Knowledge Bases for Europe's Higher Education Landscape. *Semantic Web – Interoperability, Usability, Applicability*, 4(1), 115.
- Fernández-López, M. y.-P. (1997). METHONTOLOGY: From Ontological Art Towards Ontological Engineering. *AAAI-97 Spring Symposium Series*, (págs. 24-26). Stanford University, EEUU.
- Fundación CTIC. (2013). *Public Dataset Catalogs Faceted Browser*. Recuperado el 11 de 10 de 2013, de <http://datos.fundacionctic.org/sandbox/catalog/faceted/>
- Heath, T., & Bizer, C. (2011). *Linked Data: Evolving the Web into a Global Data Space* (First ed.). Morgan & Claypool.
- Office, C. (2012). *Open Data White Paper: Unleashing the Potential*. TSO (The Stationery Office).
- Polleres, A., Scharffe, F., & Schindlauer, R. (2007). SPARQL++ for Mapping Between RDF Vocabularies. En *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS* (págs. 878-896). Springer Berlin Heidelberg.
- Poveda-Villalón, M. (2012). A Reuse-Based Lightweight Method for Developing Linked Data Ontologies and Vocabularies. En *The Semantic Web: Research and Applications* (págs. 833-837). Springer Berlin Heidelberg.
- Suárez de Figueroa Baonza, M. d. (2010). NeOn Methodology for Building Ontology. *M.C. Doctoral Thesis*. Universidad Politécnica de Madrid.
- W3C. (2008). *Best Practice Recipes for Publishing RDF Vocabularies*. Recuperado el 11 de 10 de 2013, de <http://www.w3.org/TR/swbp-vocab-pub/>
- W3C. (2008). *Cool URIs for the Semantic Web*. Recuperado el 11 de 10 de 2013, de <http://www.w3.org/TR/cooluris/>
- Zemmouchi-Ghomari, L., & Ghomari, A. R. (2013). Process of Building Reference Ontology for Higher Education. *World Congress on Engineering 2013, III*, 1595-1600.

Interdependent Components for the Development of Accessible XUL Applications for Screen Reader Users

Xabier Valencia¹, Myriam Arrue¹, Halena Rojas-Valduciel² and Lourdes Moreno²

¹*EGOKITUZ: Laboratory of HCI for Special Needs, University of the Basque Country (UPV/EHU), Informatika Fakultatea 20018, Donostia, Spain*

²*Computer Science department, Universidad Carlos III de Madrid, 28911, Leganés, Spain
{xabier.valencia, myriam.arrue}@ehu.es, halenarojas@gmail.com, lmoreno@inf.uc3m.es*

Keywords: Accessibility, User Testing, Expert Review, XUL, Mozilla Firefox Browser, Screen Reader.

Abstract: Web applications based on XUL technology have reached great development. This technology enables developers to easily create extensions and add-ons of Mozilla Firefox browser. It is essential to keep in mind accessibility in the development of such applications in order to not discriminate user groups. In this sense, standards and good practices have to be considered. Furthermore, User-Centred Design and Inclusive Design approaches should be followed as they involve users with disabilities in the development process. This paper presents an analysis of XUL accessibility guidelines created by Mozilla Foundation. An accessible XUL application has been designed and developed based on the guidelines. User testing has been conducted by two blind users revealing several important accessibility barriers. In addition, an expert review process was carried on by a blind accessibility consultant. They all used JAWS screen reader. The results obtained show that the existing guidelines conformance is not enough for ensuring accessibility of the application. There are other factors dependent on assistive technologies and user agent that have to be considered in the development of accessible XUL applications.

1 INTRODUCTION

The use of Internet has experienced a vertiginous growth in the last few years. Users access the Web employing diverse devices, modalities and technologies. Due to this diversity, inclusion approaches are necessary in order to provide full accessibility to Web contents and avoid the exclusion of some user groups.

Users with disabilities are the most affected by accessibility barriers on the Web. They access the Web using assistive technologies, for example, a screen reader that relates content in audio to the visually impaired. It is essential to develop accessible web applications to ensure appropriate assistive technology support.

Currently, research work regarding web browser functionality augmentation is gaining attention. Some examples of Mozilla Firefox add-ons are (Greasmonky, 2012), (Stylish, 2013) and (Turn off the Lights, 2013).

These augmented functionalities could be utilized by all users only if accessibility aspects are considered in their development process. Thus, a

comprehensive inclusive design paradigm for augmented browser functionalities should integrate User-Centred Design (UCD) methods and Inclusive Design approaches in addition to accessibility guidelines compliance (Lawton, 2007) (Newell, 2000).

The interest of “design for all” paradigm is rapidly increasing in the community and several efforts have been made in this way. In fact, there are several organizations concerned with web accessibility. They develop and maintain support resources for complying with accessibility standards such as guidelines. This is the case of Mozilla Foundation (XUL, 2013b). In addition, it provides extension mechanisms to augment browser functionality and develop application add-ons for Mozilla Firefox through one specific technology such as XUL (XUL, 2013a).

The objective of this paper is to analyse the appropriateness of the set of accessibility guidelines defined for XUL technology. For achieving this objective, an accessible add-on for augmenting Mozilla Firefox browser functionalities has been developed. In the development process, a

comprehensive inclusive design paradigm has been applied, including UCD approach and user testing. User testing was performed by two screen reader users. Several accessibility barriers were observed in the testing which were later on evaluated by an expert screen reader user who works as an accessibility consultant. As a result, a review of the XUL accessibility guidelines and conclusions of the development process carried on are presented.

2 OVERVIEW

2.1 XUL

XUL (XML User Interface Language) is a XML based language to create User Interfaces (UIs), in the Mozilla platform. The main goal of the language is to allow easy development of cross-platform add-on applications which run on any Mozilla integrated platform.

It separates the program logic from the user interface components, facilitating the work of the designers and programmers. This approach is also applied in languages like (QML, 2013) or (XAML, 2013).

XUL is based on existing standards such as XML, HTML, CSS, DOM and Javascript. In addition, the Cross-Platform Component Object Model (XPCOM) technology can be applied (XPCOM, 2013) when operating system functionalities are required.

2.2 Related Work

In the development processes, technological, human and legislative aspects must be considered in order to manage accessibility issues. Consequently, related work from numerous disciplines should be taken into account. In the standardization field, the W3C ought to be highlighted along with the Web Accessibility Initiative (WAI) (WAI, 2013). The Web Content Accessibility Guidelines (WCAG) 2.0 (WCAG 2.0, 2008) is one of the most important components, and is viewed as the official standard.

The ISO 9241-210:2010 standard provides a framework for following and incorporating a UCD approach into a particular context of accessibility. Following methods that integrate usability and accessibility in products design processes will ensure that users with and without disabilities could be able to access. This is the distinguishing characteristic that User Sensitive Inclusive Design (Abascal et al., 2007) has; the user with disabilities

is in mind. This work is focused on carrying out user testing technique in order to validate the accessibility included in a web application.

Several works related to XUL and accessibility has been found in the literature. These research works are related to developments of browser extensions for Mozilla Firefox. All of them are oriented to people with disabilities (Mirri et al., 2011) (Hanson et al., 2005).

Nevertheless, very few articles have been found that directly address the question of how to model accessibility according to the WCAG (Moreno et al, 2013), (Martin et al, 2010). An interesting attempt meriting particular mention is the Dante approach integrating the Web Authoring for Accessibility (WafA) ontology (Yesilada et al, 2004) (Harper and Yesilada, 2007) for the visually impaired into WSDM (Plessers et al, 2005).

2.3 XUL Accessibility Guidelines

XUL language is based on web standards so its accessibility guidelines do not differ too much from previously published web accessibility guidelines.

The XUL accessibility guidelines are divided in six different sections: Keyboard Access, Assistive Information, Display, Human Computer Interaction, Media and Custom Widgets (XUL, 2013b). Each of one has a set of checkpoints to verify. For instance, the guideline Keyboard Access defines eight checkpoints that should be considered: one related to tab order, another one to keyboard shortcuts and so on.

All the guidelines and the related checkpoints can be seen as an accessibility checklist to be considered in order to evaluate the accessibility of a developed add-on application. The XUL accessibility guidelines document states a pass/fail statement to each checkpoint which could be applied in order to elaborate a checklist easy to verify by developers at design time. Table 1 presents the checklist for evaluating XUL accessibility guidelines.

3 EXPERIMENTAL DESIGN

3.1 Object of Study

The object of study is to analyse the appropriateness of XUL Accessibility Guidelines for the development of accessible XUL-based applications.

Table 1: XUL accessibility checklist.

| | |
|-----|---|
| 1.1 | Logical tab order is provided |
| 1.2 | Keyboard functionality is provided for inaccessible features such as the column picker or added features such as column sorting |
| 1.3 | Keyboard alternatives are provided for toolbarbutton functionality |
| 1.4 | Keyboard shortcuts are present for important functionality |
| 1.5 | Context menus are triggered by the oncontextmenu event handler |
| 1.6 | All mouse operations have keyboard accessible equivalents |
| 1.7 | All scrollable elements are controllable with the keyboard |
| 1.8 | Keyboard focus is maintained and does not move unexpectedly |
| 2.1 | Alternative text is provided for meaningful images |
| 2.2 | All windows, including dialogs and wizards, have a descriptive title |
| 2.3 | Every form element has an associated label and radiobuttons are encapsulated in a groupbox |
| 3.1 | System settings are maintained |
| 3.2 | Color alone is not used to convey meaning and sufficient contrast exists between font color and background color |
| 3.3 | Visual elements and containers resize gracefully |
| 4.1 | Help documentation is provided including a description of keyboard shortcuts |
| 4.2 | Alerts are displayed using the alert scripting function or the notification box element |
| 4.3 | Interactive elements are sufficiently large and visible |
| 4.4 | Alerts are presented when the user initiates an error. The user has the opportunity and instruction to fix the error |
| 4.5 | User is informed of time limits and has control of response time when appropriate |
| 5.1 | Transcripts are provided for audio tracks |
| 5.2 | Video is captioned and a transcript is provided |
| 5.3 | User has control over animation and is warned about flashing content |
| 6.1 | Custom widgets provide accessible functionality |

3.2 Experiment Context

Following XUL accessibility guidelines, a XUL-based application has been developed. Its

accessibility has been evaluated using the XUL Accessibility Checklist (see Table 1). The developed application is an add-on for augmenting Mozilla Firefox browser functionalities.

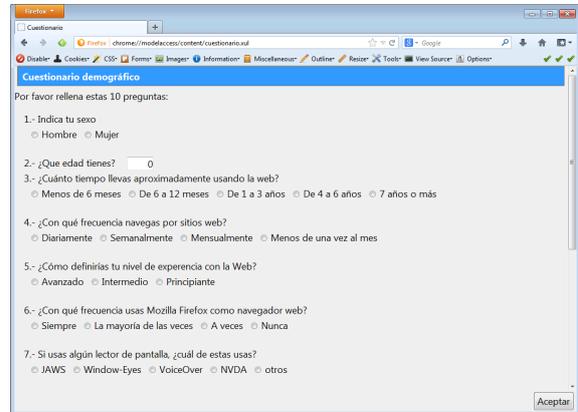


Figure 1: Demographic information form.

3.3 Sample

The add-on application for Mozilla Firefox includes several pages with different type of web content.

For this study, two web pages containing forms have been selected. The selection of such type of application was due to the following reasons: the diversity of elements included in it, importance of accessibility in order to get to each question of the forms and fill it in and the high interaction it requires from users (Lazar et al., 2007).

The questions that users are required to fill in are about demographic information, emotional aspects and issues related to the design of the visited web pages. These forms are presented to the user after some specific time interval browsing in a website. Figure 1 shows one the forms developed based on XUL.

Forms were implemented using the following XUL elements: *Window*, *Radiogroup/Radio*, *Textbox*, *Label/Description*, *Image*, *Button*, *Hbox/Vbox/Box*, *DialogHeader* and *Spacer*.

The developed forms share a similar structure. We can resume this XUL structure in the following way:

- Each form is a *Window* element which has a *Box* as a container of the form.
- The title of the form is defined with a *DialogHeader* element.
- *Description* elements have been included for providing explanations.

The input elements are one of the following: text inputs, number inputs and radio inputs. In Figure 2 an extract of a XUL document is shown.

As it can be appreciated in Figure 2, a *label* has been attached to the *radiogroup* through the *control* attribute. This mechanism ensures that assistive technologies would adequately present the form to the user.

```
<label class="question"
  value="3.- ¿Cuánto tiempo llevas aproximadamente usando la web?"
  control="uso"/>
<radiogroup class="response" orient="horizontal" id="uso">
  <radio id="menos" label="Menos de 6 meses"/>
  <radio id="seisA12" label="De 6 a 12 meses"/>
  <radio id="unoA3" label="De 1 a 3 años"/>
  <radio id="tresA6" label="De 4 a 6 años"/>
  <radio id="mas" label="7 años o más"/>
</radiogroup>
```

Figure 2: XUL extract for demographic information form.

The other input elements have been implemented similarly, ensuring that all *title* values are unique and that all *labels* were attached to the corresponding *input* element

The implemented application was verified by developers using the checklist presented in Table 1. Not all the checkpoints are relevant to the developed application as some of them are considered content not usually present in common forms. The results obtained in this initial evaluation are presented in Table 2.

All checkpoints in the checklist were fulfilled with the exception of item 4.1, the help function. It was decided not to include this function in this preliminary version. However, for upcoming development iterations, the help function will be incorporated. In conclusion, we would not expect interaction problems with the application as almost checkpoints were fulfilled. Therefore, accessibility of the application was ensured.

3.4 Participants

Two screen reader users with more than 6 years of expertise browsing the web were recruited for the user testing: a woman (User 1) and a man (User 2) whose ages were 30 and 40 respectively. User 1 considered herself as an intermediate Web browser user whereas User 2 considered himself as an advanced user.

They both use JAWS, but one of them (User 1) uses it infrequently as she prefers to use VoiceOver screen reader on Mac OS operating system. User 2 uses Windows and JAWS, but he does not usually use Firefox. Even though, both are Mozilla Firefox sporadic users.

The experimental sessions were carried out in the same lab. They were asked to bring their laptop so they used JAWS configured with their personal

preferences. The platform used was similar for both users: Windows operating system (User 1 used Windows XP and User 2 used Windows 7), JAWS 12 and Mozilla Firefox 22. Users were encouraged to report any barrier they detect when interacting with the XUL application. The sessions were recorded with a camera located behind the user in order to obtain information about the interaction. The interviews were taped with a voice recorder.

Table 2: The XUL accessibility checklist applied.

| | |
|-----|---|
| 1.1 | The tab order works correctly |
| 1.4 | Buttons have shortcuts |
| 1.6 | All actions are accessible from keyboard and mouse |
| 1.7 | The scroll can be done with the keyboard |
| 1.8 | The focus works as expected |
| 2.1 | The images have no alt text since are targeted to other users and it has nothing meaningful for blind |
| 2.2 | All windows have different titles |
| 2.3 | Labels are connected with their input element |
| 3.1 | Elements size has been set using “em” units |
| 3.2 | Elements colour have not meaning and the fonts and background has enough contrast |
| 3.3 | Flex elements has been used to avoid unexpected UI behaviours. |
| 4.1 | It is not provided in this draft version |
| 4.2 | Alerts are displayed using the alert scripting function |
| 4.3 | Form is clearly differentiated |
| 4.4 | Uncompleted form or errors are advised |

In addition, an expert evaluation was performed by an expert screen reader user who has been working as ICT accessibility consultant at least for the last five years. All the evaluation was carried on at her usual working setting, and reported her findings by email.

3.5 Procedure

First, the XUL application was installed on users’ laptop. Then, all users were asked to perform two tasks. The first one consisted on freely navigate during five minutes in a concrete website (www.discapnet.com). Then, the application presented the first form to complete. The questions in this form were related to their navigation experience. The second task consisted on a search task on the same website with a limited time interval of ten minutes. Finally, the application presented the second form containing questions related to demographic data.

4 RESULTS

4.1 User Testing

4.1.1 User 1

This user experienced several barriers when filling in the forms developed with XUL. The barriers reported by the user are the following. Besides, related XUL guidelines are indicated in each case:

- Barrier 1.1: She was unable to know which answer option was checked in the multiple-choice type questions. This occurred when the user wanted to verify that the selected answer was the correct one. She navigated with the virtual cursor of JAWS and it read the labels correctly. However, it only read the value of the option and informed that it was not checked even if it really was checked. Therefore, we had to tell her which option was selected in order to ensure that it was the desired one. (Related guideline: 2.3)
- Barrier 1.2: She experienced navigation problems. Surprisingly, when she filled in a multiple-choice question, JAWS focus was moved to the first question of the form. Then, she had to navigate to the next question from the beginning of the form. This happened even though the program focus was at the correct position. (Related guidelines: 1.1, 1.8)
- Barrier 1.3: JAWS shortcuts navigation feature did not work properly. Due to the difficulties she was having, she tried to navigate through the form using the JAWS shortcuts, like for instance, forms or headings shortcuts but it did not worked as expected. (Related guideline: 2.3)
- Barrier 1.4: Problems for clicking on a button. She was unable to find the button to continue. The button was located at the end of the form and was accessible using tab or arrows. However, JAWS did not correctly detect this element. (Related guideline: 2.3)

4.1.2 User 2

This user experienced similar problems reported by User 1 except of Barrier 1.3 (he did not try this mode of navigation). In addition, he reported other barriers:

- Barrier 2.1: Problems with text input questions. JAWS was unable to detect a text area element even if the focus was on one. Sometimes, he reported listening a label that was not the correct one. (Related guideline: 2.3)
- Barrier 2.2: Alert messages were not adequately

presented. JAWS detected the alert windows but not the containing text. Therefore, he only could hear the default sound of the alert and “OK” button but he missed the alert message. (Related guideline: 4.2)

- Barrier 2.3: Problems with numeric type inputs. Firefox adds special controls for this type of inputs. These controls are for entering the numeric value using two small buttons inside the element, one of them for increasing the value and the other for reducing it. These controls entered into conflict with his navigation controls. They are activated with keyboard arrows which were the navigation mode used by this user. Consequently, he was unable to correctly enter his age. (Related guidelines: 1.6, 2.3)
- Barrier 2.4: JAWS read not existing options in the UI. He reported us that JAWS sometimes read text elements that were not in the UI. (Related guideline: 2.3)

4.1.3 Discussion of Results

The user testing carried on indicates that there are quite accessibility barriers in the developed XUL application, even if accessibility guidelines have been considered.

Some of the detected barriers are high impact ones as users could not complete the tasks without any assistance, for instance, barrier 1.4 (Problems for clicking on a button) experienced by both users. This barrier is related to activating the submit button of the forms. Users could not get to these buttons so they could not complete the tasks on their own. These types of barriers are accessibility problems, which should be documented in the accessibility guidelines. In the group of accessibility barriers should be also included the following ones: Barrier 1.1, Barrier 2.1, Barrier 2.2 and Barrier 2.3.

Other group of barriers detected in the user testing were of moderate impact, as they do not compromise the accessibility of the application. However, they make the application less usable and users can be disappointed inducing negatively in their accessibility perception. These barriers should be erased as well for ensuring a satisfactory user interaction. For instance, Barrier 1.3 (JAWS shortcuts navigation feature did not work properly) user has other alternatives of navigation with the screen reader so this problem does not compromise the correct completion of tasks. Even though, the inexistence of this barrier makes the application more usable and user experience could be more satisfactory. In the group of usability barriers should

be also included the following ones: Barrier 1.2 and Barrier 2.4.

The detected barriers influenced negatively in the user satisfaction when interacting with the XUL application and they both needed around 15 minutes to fill in each form. Considering that each form consisted of ten short questions the time spent is not acceptable.

There are remarkable differences between the results obtained by each user. User 1 detected only two high impact barriers (Barrier 1.1 and Barrier 1.4) whereas User 2 experienced more barriers of this type (Barrier 1.1, Barrier 1.4, Barrier 2.1, Barrier 2.2 and Barrier 2.3). They both used the same version of the screen reader but the navigation strategies applied can differ a lot from user to user. In addition, our opinion is that User 1 is a more experienced user. Our observation in the experimental sessions revealed that User 1 has a wide range of knowledge about functionalities of screen readers. This observation differs from the perception of their own expertise as User 1 defines herself as an intermediate user and User 2 as an advanced one.

All in all, accessibility guidelines should consider all potential barriers independently of assistive technology version used, navigation strategies applied and user expertise level.

4.1.4 Improvements to XUL Guidelines

As can be seen most of the barriers are related to those guidelines regarding form elements, like the guideline 2.3 or the keyboard related issues 1.1 or 1.8.

Tagging labels with the corresponding control seems to be not enough. It is essential to correctly identify all questions and provide mechanisms in order to alert user and assistive technology about the existence of a list of questions or choices. Adding a new XUL element to tag the whole form would allow assistive technologies to handle better the information and also ensure the correct behaviour of the screen reader cursor.

Orientation of screen reader users would considerably improve by applying a simple good practice: informing at the beginning of the form about the total number of questions in it and numbering each question.

Regarding the keyboard, the added controls and shortcuts could create conflicts with browser controls or shortcuts as well as with assistive technologies controls. In this sense, information about the shortcuts available in the most used

assistive technologies would be helpful. This would allow a better and more efficient navigation to the user and would avoid unexpected technology behaviour.

Finally, for the alert message issues, instead of using the standard alert element, the “notificationbox” should be used. In our preliminary tests, this element seemed to work better with JAWS. However, it may cause inconveniences to other type of users such as those using magnifiers. Another alternative solution could be to apply alert functions on the active window. This issue requires more investigation to carry on.

4.2 Expert Evaluation

Due to the distinct results obtained by the users and in order to obtain a factual knowledge of the situation; a review process was conducted by an expert blind screen reader user. She is a consultant on ICT accessibility.

She was asked to conduct a test with different versions of JAWS, Firefox and operating systems. A summary of the testing can be found in Table 3. Results of the testing show the strong dependency that XUL accessibility features have with the user agent, the operating system and version of the screen reader. For instance, checkpoint 2.2 “All windows have different titles” only can be correctly detected if the user interacts with the application on a Windows 7, JAWS 13 and Firefox 23 platform. The fulfilment of checkpoint 2.2 is not so essential for user interaction (the user could complete a task even if the title of windows are not correctly presented). However, the same platform is required for checkpoint 2.3 “Labels are connected with their input element”. This checkpoint is essential if the user is supposed to access and fill in questions presented in a form.

The expert noted that Firefox generates inaccessible HTML elements for the application interface. Therefore, JAWS does not correctly read them to the user. This is due to the way JavaScript implements and interprets the DOM. Visually, the element appears in the interface, but it is as if the element was non-existent in the DOM. The screen reader just ignores it.

In conclusion, the latest version of Firefox and JAWS (at least versions JAWS 13 and Firefox 23) seems to be the best combination for using the developed XUL application. Nevertheless, it is unusual that users have the latest versions of screen readers. The same occurs with browsers. Most of screen reader users use outdated browser versions

since the cursor mode of JAWS does not work with the latest versions. Sometimes this mode is necessary in order to access user interface elements that cannot be read as usually.

Table 3: Correspondence of XUL accessibility checklist with the expert review results.

| | |
|-----|---|
| 1.1 | Only in Firefox 23. |
| 1.4 | Better using Firefox 23 |
| 1.6 | Only using Firefox 23 |
| 1.7 | Only using Firefox 23 |
| 1.8 | JAWS 13 and Firefox 23 |
| 2.1 | When running the extension on Windows 7, using Firefox 23 |
| 2.2 | Windows 7, JAWS 13 and Firefox 23 |
| 2.3 | Only using JAWS 13 and Firefox 23 |
| 3.1 | Good feature for low vision using a magnifier |
| 3.2 | Good feature for low vision |
| 3.3 | Better using Firefox 23 and JAWS 13, JAWS 14 |
| 4.1 | - |
| 4.2 | Better using Firefox 23 |
| 4.3 | When refreshing the virtual buffer of JAWS, refresh the page or when the form is being read automatically |
| 4.4 | Only using Firefox 23 JAWS 13 and Windows 7 |

Some suggestions for improving the current version of the developed XUL application were indicated by the expert:

- The text for the questions should be defined as a header element.
- The multiple-choice questions should define the possible answers as a list element.
- Users should be provided by an input text element for introducing the answer. This suggestion would increase the form response time, therefore it would be less efficient and usable. However, it would be accessible for screen reader users.

Implementing these suggestions would overcome some of the most significant problems that users will have probably to face to. Mainly when they do not use the latest versions of browser and screen reader.

5 LESSONS LEARNED

From the results obtained in user testing and expert evaluation, it is clear that there are extremely important accessibility barriers in the developed XUL application. The user testing can bring up problems that have not been detected previously.

Findings reported in section 4 also highlight the importance of testing applications with different versions of Firefox and screen reader. There are too

many differences between versions and possible combinations. Each combination has its strengths and weaknesses. Using the latest versions of each one seems to be the best solution. Even though, many people do not update their software, principally, due to the price of these updates. But sometimes there are some functionalities that the user is used to, that he cannot leave behind. For these reasons, it is crucial to ensure that the developed applications can be used in the wider range of versions as possible.

User agent developers should consider the accessibility barriers detected in this work. Universal access to existing augmented browser applications can be guaranteed only if inclusion design paradigms are adequately defined and applied.

In the mean time, a transitory solution could be to transform the application into a browser XUL element to display the forms coded in HTML inside the XUL code. As HTML accessibility issues have been more analysed and considered in the last decades. A great amount of documentation, tools and methods exist for making accessible HTML code. It would avoid the creation of the keyboard scripts, because the keyboard behaviour would work as expected and also reduces the workload. But the main advantage is that it would make possible a higher compatibility between different Firefox versions and screen readers.

Anyway, XUL accessibility guidelines should be reviewed in order to update issues regarding keyboard navigation, assistive technologies compatibility, forms elements tagging, etc. Not only to ensure that all elements are accessible but guarantee also the HTML-like behaviour so users do not get confused or disoriented during the interaction.

6 CONCLUSIONS

Web access to all users should be ensured, including people with disabilities who use assistive technology to access ICTs. Many organizations are concerned about this issue, and they work towards accessibility compliance. This is the case of the Mozilla Foundation that provides accessibility guidelines to apply when developers use their technologies like XUL.

This paper presents a study of XUL and its accessibility guidelines. We have developed an application for augmenting browser functionalities in XUL. The development process considered UCD approach with the aim of creating an accessible

application with form content type. User testing with 2 legally blind users was carried on in addition to accessibility guidelines conformance evaluation. The developed application seemed to be accessible according to XUL accessibility guidelines. However, the results gathered in the user testing indicated important accessibility barriers. Some of the detected barriers make the application not operable for screen reader users.

An expert evaluation has also been considered in this paper. A screen reader user with more than five years of experience as a consultant on ICT accessibility has conducted the review. The results reveal a strong dependence between platform used (versions of user agent, operating system and screen reader) and the accessibility barriers experienced by users.

The findings of this paper should be considered in next versions of XUL accessibility guidelines. Some of them could be included as guidelines whereas others could be considered as best practices.

This research was oriented to screen reader users and XUL applications containing forms. In the near future, there is a need of performing evaluation studies with more content type, other groups of users and other assistive technologies. Therefore, future work will be motivated to the evaluation and improvement of other XUL accessibility checkpoints not considered in this work. This research work will lead to ensure universal access to Mozilla Firefox add-on applications.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the Spanish Ministry of Science and Innovation through Project ModelAccess (TIN2010-15549) and the MULTIMEDICA project (TIN2010-20644-C03-01).

EGOKITUZ is supported by the Department of Education, Universities and Research of the Basque Government (Eusko Jaurlaritz/Gobierno Vasco) through Grant# IT-395-10.

REFERENCES

Abascal, J and Azevedo, L. *Fundamentals of Inclusive HCI. Design.* (2007) Universal Access in Human Computer Interaction, 4th International Conference on Universal Access in Human-Computer Interaction, UAHCI 2007, Held as Part of HCI International 2007, Beijing, China, July 22-27.

Greasmonky (2012) http://wiki.greasespot.net/Main_Page.

Hanson, V. L., Brezin, J., Crayne, S., Keates, S., Kjeldsen, R., Richards, J. T., Swart, C., & Trewin, S. (2005). *Improving Web accessibility through an enhanced open-source browser.* IBM Systems Journal, 44 (3), 573 - 588.

Hanson V. L., Richards. J. T., and Swart. C., (2008) *Browser augmentation*, Harper S. and Yesilada Y, Web Accessibility, Springer London, pp. 215-229.

Harper, S. and Yesilada, Y., (2007). *Web Authoring for Accessibility (WafA).* Web Semantics: Science, Services and Agents on the World Wide Web. 5, 3, pp. 175-179.

Lawton S, H. (2007) *Just Ask: Integrating Accessibility Throughout Design.* Madison.: ET\Lawton, available at www.uiAccess.com/justask/

Lazar J, Allen A, Kleinman J, Malarkey C. (2007) *What Frustrates Screen Reader Users on the Web: A Study of 100 Blind Users*, International Journal of human-computer interaction, Taylor & Francis, 22 (3), pp. 247-269.

Martín, A, Rossi, G, Cechich, A and Gordillo, S. (2010) *Engineering Accessible Web Applications. An Aspect-Oriented Approach.* World Wide Web, Springer US 13 (4), pp. 419-440,

Mirri, S, Salomoni, P, and Prandi, C. 2011. *Augment browsing and standard profiling for enhancing web accessibility.* In Proceedings of the International Cross-Disciplinary Conference on Web Accessibility. ACM, New York, NY, USA, Article 5 , 10 pages.

Moreno L, Valverde F, Martínez P, Pastor O. (2013). *Supporting accessibility in Web engineering methods: a methodological approach*, January, 2013, Journal of Web Engineering , RINTON PRESS, INC, ISSN: 12 (3&4), pp. 1540-9589.

Newell, A.F.; Gregor, P. (2000) *User Sensitive Inclusive Design: in search of a new paradigm.* Proceedings on the 2000 conference on Universal Usability, pp. 39-44.

Plessers P, Casteleyn S, Yesilada Y, De Troyer O, Stevens R, Harper S, and Goble C (2005) *Accessibility: A Web Engineering Approach.* In Proceedings of the 14th International Conference on World Wide Web (WWW '05) ACM, New York, NY, USA, pp. 353-362.

QML (2013) <http://qt-project.org/doc/qt-5.0/qtquick/qtquick-applicationdevelopers.html>.

Stylish <http://userstyles.org/>

Turn off the Lights (2013) <http://www.stefanvd.net/project/turnoffthelights.htm>.

WAI (2013) <http://www.w3.org/WAI/>

WCAG 2.0 (2008). <http://www.w3.org/TR/WCAG20/>

XAML (2013) <http://msdn.microsoft.com/en-us/library/ms752059.aspx>.

XPCOM (2013) <https://developer.mozilla.org/en-US/docs/XPCOM>.

XUL (2013a) <https://developer.mozilla.org/en-US/docs/XUL>.

XUL Accessibility Guidelines (2013b) https://developer.mozilla.org/en-US/docs/XUL_accessibility_guidelines.

Yesilada, Y., Harper, S., Goble, C. and Stevens. (2004) R. *Dante annotation and transformation of web pages for visually impaired users*. In The Thirteenth International World Wide Web Conference. ACM, New York, NY, USA, pp. 490-491.

Hypermodal

Dynamic Media Synchronization and Coordination between WebRTC Browsers

Li Li¹, Wen Chen², Zhe Wang³ and Wu Chou¹

¹Shannon Lab, Huawei Technologies, Bridgewater, New Jersey, U.S.A.

²Contractor, Global Infotech Corporation, Kearny, New Jersey, U.S.A.

³CS Dept., Rutgers University, Piscataway, New Jersey, U.S.A.

{li.nj.li, wu.chou}@huawei.com, chenwen47@yahoo.com, zhewang@cs.rutgers.edu

Keywords: Temporal Linkage, Synchronization Tree, RDF, Media Fragments URI, WebRTC, REST API.

Abstract: This paper describes a Web based real-time collaboration system, Hypermodal, based on the concept of temporal linkage between resources. The system allows the users to construct, manipulate and exchange temporal linkages organized as synchronization trees. The temporal linkage is defined by RDF <sync> predicate based on a novel use of Media Fragments URI and permits on-the-fly tree updates while the resources in the tree are playing. We propose RDF <mirror> predicate and a new protocol to correlate and initialize distributed synchronization trees without requiring clock synchronization. Moreover, we develop a new REST API optimized for efficient tree updates and navigations based on super nodes. The preliminary test results on a prototype system show the approach is feasible and promising.

1 INTRODUCTION

The true power of the Web is to organize distributed information in an unconstrained way through hypertext (Berners-Lee, 2000). With the vast amount of multimedia resources on the Web and the advent of WebRTC, there is an acute need to be able to link these real-time multimedia resources in an accurate and meaningful way.

The continuous nature of multimedia resources makes it insufficient to link them the way we link discrete documents or images. What we need is a new type of temporal linkage that can link intervals, regions or objects within multimedia resources. The application domains of such technology are wide open. We can link a person in a video stream to his home page so that the conference participants can find more about him without asking. When discussing a trip to Barcelona Spain, we can link the conversation to a Google map, a Wikipedia page, and a public transportation page about the city. Users of MOOCs websites can link part of an online video lecture to relevant segments of another video during a live discussion such that students can learn the same concepts from different professors. The agents that link the resources can also be machine programs, such as Speech Recognition, Machine Translation, or Face Tracking and Detection

engines. For example, a moderator can schedule conference topics and a topic search engine can link resources relevant to the topics on time into the conference.

Temporal linkages can even link discrete resources without a temporal dimension, by treating them as continuous resources whose content does not change in small time scale. They can also link abstract resources that have a temporal dimension but no intrinsic content, such as a session. This generalization gives us the ability to temporarily link any types of resources in anywhere in a uniform way.

This paper describes a real-time collaboration system Hypermodal based on the concept of temporal linkage. The system allows the users to construct, manipulate and exchange temporal linkages in a meaningful way in real-time. Our goal is to create a mutual feedback loop between the system and the Web: any Web resources can be linked to the system and the links created by the system become part of the Web. To achieve this goal, we use as many standards as possible such that the components processing the temporal linkages can be developed independently but fully interoperate. Under this guideline, we address the following research issues in this paper.

If not constrained, the temporal linkages can

form an arbitrary directed graph which is difficult to manage for both users and machines. To solve this problem, we propose RDF `<sync>` predicate to define a generic temporal linkage based on a novel use of W3C Media Fragments URI standard (Troncy, 2012), such that we can construct synchronization trees, instead of directed graphs, to represent temporal linkages.

During a collaboration session, the synchronization trees are not static but are constructed incrementally and may change at any time when the resources in the tree are playing. To support on-the-fly updates, we develop a mechanism to play individual `<sync>` linkages without interrupting the resources in play.

In our system, whenever a user modifies a synchronization tree, the modification may propagate to other remote trees so all users have the same view. However, WebRTC Web browsers assign different URIs to the same multimedia resource, making it impossible to exchange `<sync>` linkages between synchronization trees that share the same resource. To address this problem, we propose RDF `<mirror>` predicate and a new protocol to correlate and initialize distributed synchronization trees without requiring clock synchronization between browsers and servers.

The components in our system need a protocol to communicate their updates to the synchronization trees. However, current RDF query and update languages are not designed for efficient updates to synchronization trees. To address this problem, we develop a novel REST API to navigate and update synchronization trees based on the concept of super node.

The rest of this paper is organized as follows. Section 2 surveys the related work. Section 3 defines the synchronization tree based on the `<sync>` and `<mirror>` relations. Section 4 describes how to initialize synchronization trees using WebRTC call control protocol. Section 5 introduces the REST API for managing the synchronization. Section 6 describes our prototype implementation and experimental results, and we conclude the paper with Section 7.

2 RELATED WORK

The approach described in this paper is different from the screen/application sharing offered by many Web conference systems. In screen sharing, shared content is read-only to all users except the owner, whereas in our approach, shared resources are

interactive to all users. Screen sharing requires more network bandwidth to send the encoded video than the REST API to coordinate distributed synchronization trees. For example, Skype (Skype, 2013) requires at least 128 kbits/s for screen sharing, whereas we estimate the bandwidth required by our REST API is lower than 5 kbits/s. Furthermore, screen sharing creates a loophole for the Same Origin Policy (Rescorla, 2013), whereas our approach enforces this policy.

WebRTC (Bergkvist, 2013) is an ongoing joint effort between W3C and IETF to develop Web (JavaScript) and Internet (codec and protocol) standards to enable P2P real-time communication between Web browsers without any external plug-ins. Several open source Web browsers, including Chrome (WebRTC Chrome) and Firefox (WebRTC Firefox), already support some WebRTC functions and demonstrate certain degree of interoperability.

SMIL (Bulterman, 2008) is a XML dialect to express temporal synchronization between multimedia resources. However, SMIL does not support dynamic media synchronization. Once a SMIL document begins to play, we cannot modify the document to add new media or change the synchronization relations between existing media.

Nixon (Nixon, 2013) describes a research agenda for Linked Media, inspired by the Linked Data initiative, where the main approach is to link multimedia based on the conceptual relations between the fragments of the multimedia. Li et al (Li, 2012) describes Synote, a system to interlink multimedia fragments based on RDF and Media Fragments URI. Oehme et al (Oehme et al, 2013) describes a system to link a video to Web resources and overlay the resources on top of the video. Mozilla Popcorn Maker (Popcorn Maker) is an open source JavaScript library that allows a user to create multimedia presentations by layering Web information, e.g. Google map or Wikipedia page, at different intervals and regions of an online video.

However, all these approaches are for single user presentations or based on ad-hoc languages, not for multi-user and on-the-fly resource synchronization based on temporal linkage.

3 SYNCHRONIZATION TREE

Figure 1 illustrates the smallest Hypermodal system with two Web browsers connected by a Web server, where the synchronization trees are stored in the browsers and the `<mirror>` linkages are stored on the server.

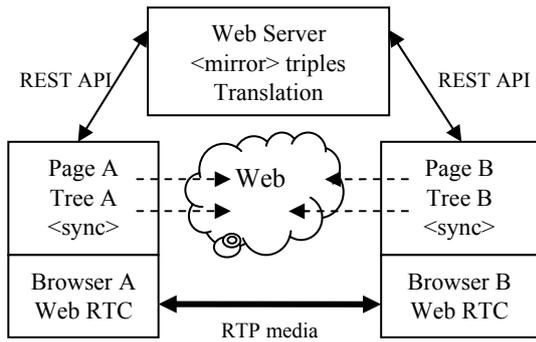


Figure 1: Basic Hypermodal system architecture.

A synchronization tree consists of nodes that define the intervals of resources based on Media Fragments URI, and edges that define temporal linkages between the nodes based on our RDF <sync> predicate, as illustrated in Figure 2, where the nodes are rendered as horizontal lines, whose lengths indicate the intervals, and the edges as vertical arrows whose positions indicate synchronization points. Both trees are rooted at the same session resource identified by URI0. The <mirror> linkages that link the same resources shared between the trees are rendered as the curved arrows.

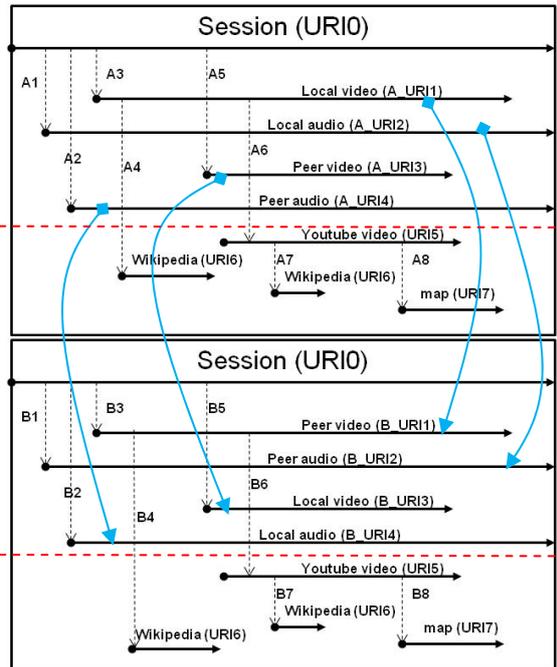


Figure 2: Correlations between tree A (top) and tree B (bottom).

Synchronization tree does not constrain the resources in any way as long as they can be identified by Media Fragments URI. For example,

we can even synchronize two different intervals of the same video. If we treat the unique resources, not the intervals of the resources, as nodes, they can form a directed graph linked by the <sync> linkage. However, a tree is a more accurate representation of <sync> linkages, because different intervals of the same resource can be updated independently as distinct resources. For instance in Figure 2, the same Wikipedia page (URI6) occurs as two nodes in two <sync> linkages (A4 and A7). When one linkage is changed, the other one is not affected. But if the page is changed, it will be reflected on both linkages.

Synchronization trees are constructed dynamically when the resources in the tree are playing: 1) new edges can be added anywhere by inserting RDF <sync> triples (edges); 2) the intervals and synchronization points of a resource can be changed by modifying the Media Fragments URI (nodes). This process is detailed in Sections 4 and 5.

3.1 The <Sync> Predicate

We propose <sync> RDF predicate whose subject and object are Media Fragments URIs. If x_s, x_e, y_s, y_e are nonnegative integers that denote the start and end time of a resource in second, then the canonical triple:

$\langle \text{URI}_X\#t=x_s, x_e \rangle \langle \text{sync} \rangle \langle \text{URI}_Y\#t=y_s, y_e \rangle$

instructs the user agent to play the interval $[y_s, y_e)$ of resource URI_Y within the interval $[x_s, x_e)$ of resource URI_X . For example, the following triple:

$\langle \text{A_URI1}\#t=50 \rangle \langle \text{sync} \rangle \langle \text{URI5}\#t=30, 120 \rangle$

plays the interval $[30, 120)$ of resource URI5 when resource A_URI1 reaches the 50th second and before it ends.

The <sync> linkage assigns different meanings to Media Fragments URI based its role in a <sync> triple: the subject interval $[x_s, x_e)$ defines synchronization points on URI_X , not a new resource extracted from URI_X , whereas the object interval $[y_s, y_e)$ defines a new resource extracted from URI_Y . This is why we can simultaneously attach many resources to intervals of URI_X while URI_X is playing, but at the same time maintain the tree structure by breaking URI_Y into independent portions.

The canonical <sync> triple requires the user agents to know the absolute play time of resources. This is difficult in distributed system when the machine clocks are not synchronized. To address this problem, we introduce two extensions to Media Fragments URI: relative delay and event

synchronizations. If d and r are positive numbers, then the relative delay triple:

$\langle \text{URI}_X\#t=+d,+r \rangle \langle \text{sync} \rangle \langle \text{URI}_Y\#t=y_s,y_e \rangle$

plays an interval of URI_Y within the interval $[\text{now}+d, \text{now}+d+r]$ of resource URI_X , where now denotes the normal play time of resource URI_X when the triple is to be played.

Event synchronization is borrowed from SMIL (Bulterman 2008). If $E1$ and $E2$ denote events generated by resource URI_X , then the triple:

$\langle \text{URI}_X\#t=E1+d,E2+r \rangle \langle \text{sync} \rangle \langle \text{URI}_Y\#t=y_s,y_e \rangle$

defines the resource play interval based on the time of the events. For example, to automatically display the PowerPoint at URI_Y when topic $t1$ is being discussed in the session that generates event $t1_s$ when $t1$ starts and event $t1_e$ when the topic ends, we could use the following triple:

$\langle \text{session}\#t=t1_s,t1_e \rangle \langle \text{sync} \rangle \langle \text{URI}_Y \rangle$

To play a canonical $\langle \text{sync} \rangle$ triple while the subject resource is playing, we use the process depicted in Figure 3, which takes the current subject play time and a $\langle \text{sync} \rangle$ triple as input, calculates the actual play time of the object resource, and produces a task as output. A $\langle \text{sync} \rangle$ triple can specify any start and end times, and it can be played at any time, but the process makes sure the actual play interval of the object resource is always within the current play time and the specified end time of the subject resource. If this is impossible, the object resource will not be played.

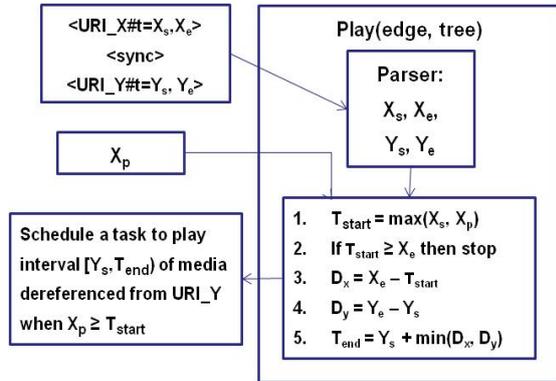


Figure 3: Process to play a $\langle \text{sync} \rangle$ triple.

To play a relative delay triple, the process translates $+d$ to absolute play time $\text{now}+d$. To play an event triple, the process translates $E+d$ to absolute play time $\text{time}(E)+d$. For example, if event $t1_s$ happens at the 600th second, and $t1_e$ happens at the 1200th second, then $\langle \text{session}\#t=t1_s,t1_e \rangle$ is translated to $\langle \text{session}\#t=600,1200 \rangle$.

The play process automatically compensates the network delay by adjusting the play interval. To illustrate this effect, suppose browser B sends browser A the $\langle \text{sync} \rangle$ triple: $\langle \text{URI5}\#t=20,30 \rangle \langle \text{sync} \rangle \langle \text{URI6} \rangle$ when the current play time of URI5 at browser A is 19. When browser A receives the triple in 2 seconds, the current play time of URI5 has advanced to $21=19+2$. Browser A will then play URI6 only for 9 seconds. This approach never rewinds a subject resource in play although it may abbreviate the play intervals of object resources. It is a reasonable approach if the network delay is relatively small compared to the play intervals. An alternative approach outlined in (Pan, 2012) is to “rewind” URI5 back to 20 to play URI6 for 10 seconds. However, such approach will not work if there are multiple conflicting “rewind” actions.

3.2 The $\langle \text{Mirror} \rangle$ Predicate

We propose $\langle \text{mirror} \rangle$ RDF predicate whose subject and object identify resources from the same source, as illustrated in Figure 4.

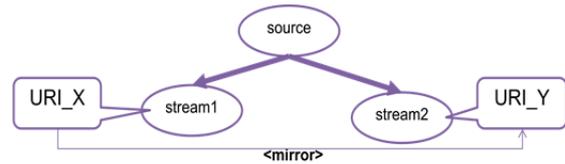


Figure 4: Resource model for $\langle \text{mirror} \rangle$ linkage.

For example, $\langle A_URI1 \rangle \langle \text{mirror} \rangle \langle B_URI1 \rangle$ indicates the two URIs in Figure 2 identify two resources whose streams come from the same video camera. The $\langle \text{mirror} \rangle$ linkage is not equivalent to RTP SSRC (Perkins, 2008), because two resources in the $\langle \text{mirror} \rangle$ linkage can have different SSRC values. For example, a media device may mix a video stream1 with an advertisement stream2 to create a picture-in-picture stream3. Although stream1 and stream3 have different SSRC, they are still regarded as the same by users because their main contents are the same.

The $\langle \text{mirror} \rangle$ closures can be computed based on the following properties:

1. Commutative: $X \langle \text{mirror} \rangle Y \Rightarrow Y \langle \text{mirror} \rangle X$
2. Transitive: $X \langle \text{mirror} \rangle Y, Y \langle \text{mirror} \rangle Z \Rightarrow X \langle \text{mirror} \rangle Z$

With these closures, the translation between $\langle \text{sync} \rangle$ triples is straightforward as shown in Figure 5.

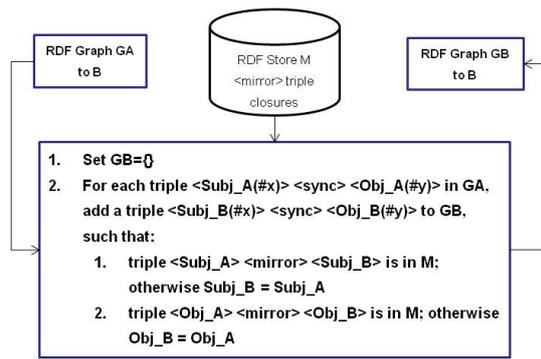


Figure 5: translation based on <mirror> closures.

4 TREE INITIALIZATION

When a user joins a collaboration session, the synchronization tree is initialized automatically by the system with the session and whatever media streams the user chooses to send and receive. To avoid performance overhead, the best approach is to embed the tree initialization protocol within the regular session establishment protocol. Figure 6 illustrates this technique using WebRTC offer/answer protocol (Rosenberg, 2002) with two additional messages ack and ok. The server sends the browsers the session URI and play time in regular answer (t_1 at step 6) and ack (t_2 at step 8) messages so that browsers can attach local media resources to the session at the correct time. The browsers send the server the correlation relations in ack (step 7) and ok (step 11) messages so that the server can derive the <mirror> relations between local URIs from the correlations. The correlation is established from media stream identifier (msid) maintained by WebRTC API.

When a browser wants to attach a resource to the session, it must know the current session time. However, the session time is maintained by the server clock, whose rate is unknown to the browser. One solution is to synchronize the clocks of the browsers and the server. But this requires additional protocol stack (e.g. NTP), which is often not available on the browsers and servers. This paper proposes two alternative approaches without requiring any dedicated time synchronization protocol.

In the *server-based* approach, the browsers use relative delay URI and let the server to figure out the session play time. For example, browser A can send <URI0#t=+0> <sync> <A_URI1> to the server, which calculates the session play time t_x by: $t_x = \text{now} - d(S, X)$ according to the current session

time now and the network delay $d(S, X) \geq 0$ between the browser and the server. The advantage of this approach is that the browser does not need to know the session time maintained by the server. The constraint is that the server needs to store the synchronization trees for the browsers.

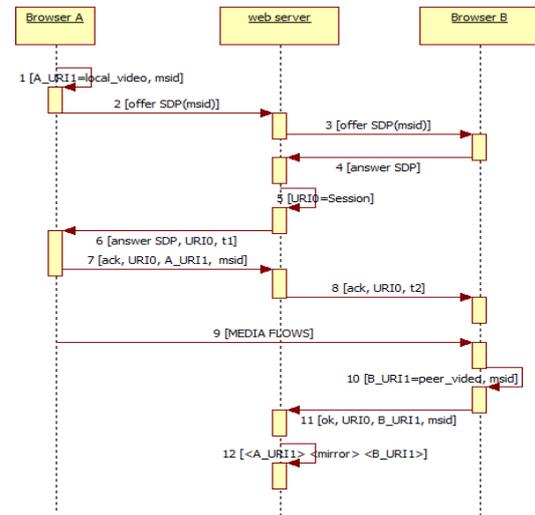


Figure 6: Tree initialization and correlation process.

In the *client-based* approach, each browser estimates the current session play time using its own clock. When a browser first receives the session time t_0 at its local time t_1 , it records these numbers. When it sends a <sync> triple at time $t_2 \geq t_1$, it estimates the current session time t_x by: $t_x = t_0 + t_2 - t_1$ and uses t_x in the <sync> triple. The advantage of this approach is that server does not have to store the synchronization trees. The disadvantage is that the estimated session time may be inaccurate when the client clock differs from the server clock.

5 SYNCHRONIZATION TREE REST API

In our system, users can frequently add <sync> triples, update the triples, or delete a triple. Users may also navigate and explore synchronization trees. Different user agents may accept different formats of a synchronization tree, e.g. XML, JSON or RDF Turtle (Manola, 2007). Servers and user agents need to cache and store the trees at different locations and formats. These use cases led us to choose REST API to encapsulate the synchronization trees.

There are several approaches (Gearon, 2013, Sesame REST API, Berners-Lee 2001, Tummarello 2007) to update RDF graphs using REST API (ref).

However, these approaches treat the entire RDF repository (i.e. a synchronization tree in our case) as a resource. To locate a RDF triple in the repository, a client must specify the subject, predicate and object of the triple. However, in a concurrent system, a client’s knowledge about a triple will be out of date if other clients have changed the triple.

To address this problem and to reduce message size, our REST API treats each <sync> triple as a resource and assigns it a unique URI that does not change, which can be easily achieved because many RDF packages such as Jena (Jena) assigns unique internal identifiers to RDF triples. With this URI, a client can locate any <sync> triple without specifying its current state. This idea is illustrated in Figure 7 where each rectangle enclosing a <sync> triple represents a REST resource with a unique URI. For example, the triple: <URI_0#T01> <sync> <URI_X#TX0> is a REST resource identified by URI_triple1.

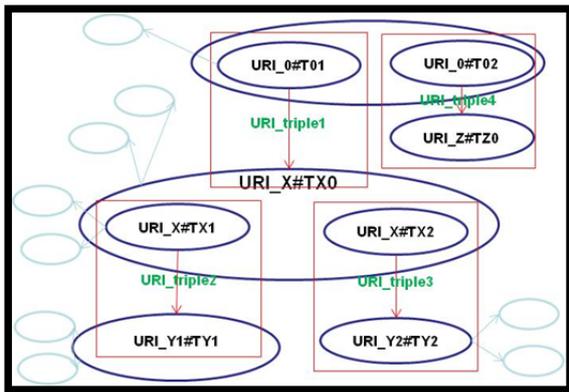


Figure 7: Resource model of synchronization tree.

However, these REST resources are not connected as the rectangles do not intersect. To address this problem, we introduce the concept of super node, represented by ovals that enclose the objects and subjects of different <sync> triples which share the same base URI (URI without fragment). For example, the super node URI_X contains URI_X#TX0 of URI_triple1, URI_X#TX1 of URI_triple2, and URI_X#TX2 of URI_triple3. Because URI_X#TX1 and URI_X#TX2 are the children of URI_X#TX0, we map these relations to the corresponding REST resources to connect them as follows:

```
<URI_triple1> <child> <URI_triple2>
<URI_triple1> <child> <URI_triple3>
```

At the top of Figure 7 is a root super node that contains a set of “sibling” URIs: URI_0#T01 of

URI_triple1 and URI_0#T02 of URI_triple4. Similarly, we map this relation to the REST resources to connect them:

```
<URI_triple1> <sibling> <URI_triple4>
```

The members of a super node are updated whenever a new <sync> triple X is attached to an existing <sync> triple Y by the following rule: assert relation: <Y> <child> <X> if the subject of X has the same base URI as the object of Y. For new triple without parents, the REST API attaches them to an internal subject, so they can be checked for <sibling> relations.

5.1 Create Triple

Figure 8 shows POST request and response to add new triples to a tree resource identified by URI_tree, or to attach new <sync> triples to an existing one, by replacing URI_tree with the URI to the triple.

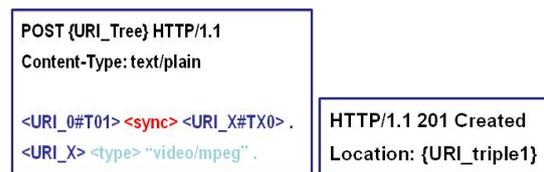


Figure 8: create a <sync> triple.

5.2 Retrieve Triple

Figure 9 shows the GET request and response that returns the requested triple and its neighbors connected by super nodes.

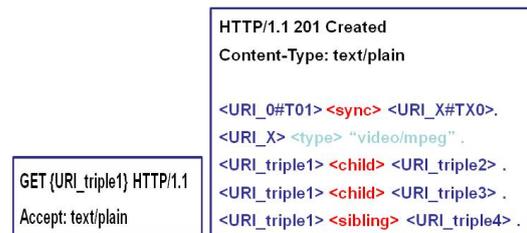


Figure 9: retrieve a <sync> triple.

5.3 Update and Delete Triple

Figure 10 shows three ways to update a <sync> triple: 1) subject interval; 2) object interval; or 3) both subject and object intervals.

We use DELETE message to a <sync> resource to delete it, which will stop its playback and remove all its child <sync> triples.

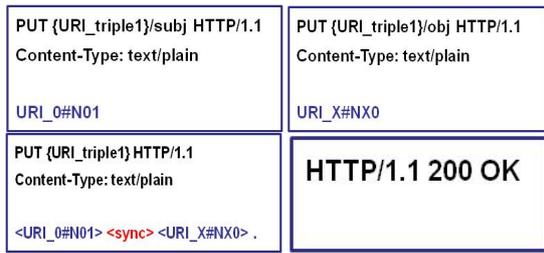


Figure 10: 3 update a <sync> triple.

6 PROTOTYPE SYSTEM

A prototype Hypermodal system was implemented based on the open source Mozilla Popcorn Maker engine (Popcorn Maker) and Jena RDF package (Jena).

Mozilla Popcorn Maker engine is implemented in JavaScript and runs in Web browsers. The engine allows a user to layer multimedia resources, such as YouTube video, Google map and Twitter stream, on a time axis. Users can change the start time and duration of the layered media during playback, and control the playback of these media with start, stop, seek, pause and resume actions.

We integrated a WebRTC call control module into Popcorn Maker so that users can make audio/video calls through the Web server. During the call negotiation, the Web server and the Web browsers initialize the synchronization trees as described in Section 4. We modified the Popcorn Maker to detect relevant events at one Popcorn Maker instance, translate these events to the REST API messages, and send these messages over WebSocket to the Web server as described in Section 5. The Web server will translate the messages based on <mirror> relations and broadcast them to the Popcorn Maker instances in other browsers.

Two screenshots of the modified Popcorn Maker interface are shown in Figure 11 for Alice (top) and Bob (bottom). Here Alice added a Google map beneath her video and the map is displayed on Bob's screen under Alice's video. Similarly, Bob added a recorded video to his video, and this video is shown on Alice's screen below Bob. Other Web resources, including Wikipedia page, Twitter page, chat window and images, can be added to the synchronization trees by the users as well.

We tested the performance of the system at the Web browsers (Lenovo Thinkpad 420 with Intel Core i5 2520M 2.50GHz (Dual Core) and 4.0GB RAM, 32-bit Windows 7 Professional) and the Web

server (Dell OptiPlex 990 Mini Tower with Intel® Core™ i7 2600 Processor (3.4GHz, 8M), 16GB RAM, 64-bit Windows® 7 Professional) in a LAN environment, when users add new <sync> triples or modify them. For each operation, the following 4 time measurements were recorded.

1. BT: browser translates UI action to RDF triples and sends them in REST API request.
2. SP: the server parses the triples in request and stores them in Jena RDF models.
3. ST: the server translates the request triples and broadcasts them to other browsers.
4. BR: round-trip time at the browser from sending REST API request to receiving response.

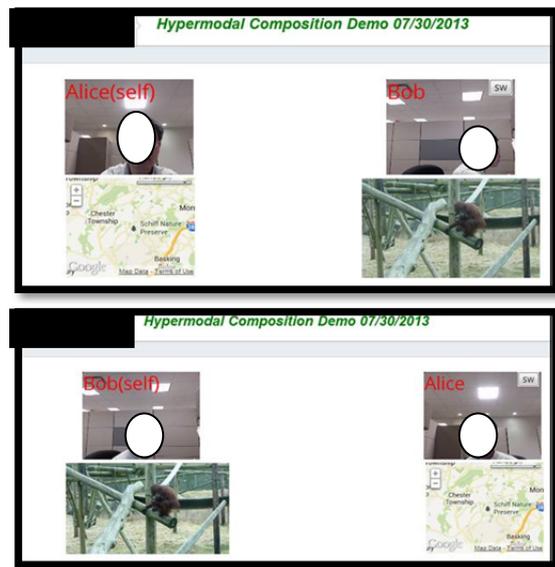


Figure 11: Screenshots of Hypermodal prototype system.

The following tables summarize the results (in millisecond) for adding the triples (top) and updating the triples (bottom) respectively, each averaged over 20 runs.

Table 1: Task time for adding and updating triples.

| Time/task | BT | SP | ST | BR |
|-----------|-------|------|------|-------|
| mean | 47.55 | 0.66 | 0.10 | 18.45 |
| std | 7.98 | 0.34 | 0.01 | 1.93 |

| Time/task | BT | SP | ST | BR |
|-----------|-------|------|------|------|
| mean | 14.05 | 0.21 | 0.24 | 6.9 |
| std | 1.93 | 0.04 | 0.05 | 0.85 |

These experimental results indicated that the proposed approach is feasible and promising since the total server processing time is less than 1 ms and

the total round-trip delay at browsers is 66 ms for adding triples and 21 ms for updating triples.

7 CONCLUSIONS

The contributions of this paper are summarized as follows.

1. A synchronization tree model based on temporal linkage defined by RDF <sync> predicate to allow dynamic modifications to the tree while the resources in the tree are playing.
2. A RDF <mirror> predicate and a new protocol to correlate and initialize distributed synchronization trees so that updates to one tree can be correctly translated to another tree without clock synchronization.
3. A novel REST API to support efficient updates on synchronization trees by treating <sync> triples as REST resources and connect them through super nodes.

For future work, we plan to extend the temporal linkage to spatial regions and objects in resources, study multimedia resource cache mechanisms for efficient constructions of synchronization trees, and security mechanisms to prevent unauthorized and malicious updates to synchronization trees, and apply the described Hypermodal system to more complex real-time collaboration applications.

REFERENCES

- Bergkvist, A. et al (ed): WebRTC 1.0: Real-time Communication Between Browsers, W3C Editor's Draft 30 August 2013, <http://dev.w3.org/2011/webrtc/editor/webrtc.html>, Last Access: October 10, 2013.
- Berners-Lee, T.: Weaving the Web, Harper, 2000.
- Berners-Lee, T. et al: Delta: an ontology for the distribution of differences between RDF graphs, 2001, <http://www.w3.org/DesignIssues/Diff>, Last Access: October 10, 2013.
- Bulterman D. et al (ed): Synchronized Multimedia Integration Language (SMIL 3.0), W3C Recommendation 01 December 2008, <http://www.w3.org/TR/SMIL3/>, Last Access: October 10, 2013.
- Gearon, P. et al (ed): SPARQL 1.1 Update, W3C Recommendation 21 March 2013, <http://www.w3.org/TR/sparql11-update/>, Last Access: October 10, 2013.
- Jena: <http://jena.apache.org/>, Last Access: October 10, 2013.
- Li, Y. et al: Synote: Weaving Media Fragments and Linked Data, LDOW2012, April 16, 2012, Lyon, France, <http://events.linkeddata.org/ldow2012/papers/ldow2012-paper-01.pdf>, Last Access: October 10, 2013.
- Manola, F. et al (ed): RDF Primer — Turtle version, <http://www.w3.org/2007/02/turtle/primer/>, Last Access: October 10, 2013.
- Nixon, L. J. B.: The Importance of Linked Media to the Future Web, WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil, <http://www2013.wwwconference.org/companion/p455.pdf>, Last Access: October 10, 2013.
- Oehme, P. et al: The Chrooma+ Approach to Enrich Video Content using HTML5, WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil, pages 479–480.
- Pan, J., Li, L., Chou, W.: Real-Time Collaborative Video Watching on Mobile Devices with REST Services, 2012 Third FTRA International Conference on Mobile, Ubiquitous, and Intelligent Computing, pages 29–34, Vancouver, Canada, June 26–28, 2012.
- Perkins, P.: *RTP, Audio and Video for the Internet*, Addison-Wesley, 2008.
- Popcorn Maker: <https://popcorn.webmaker.org/>, Last Access: October 10, 2013.
- Rescorla, E.: Notes on security for browser-based screen/application sharing, March 11, 2013, <http://lists.w3.org/Archives/Public/public-webrtc/2013Mar/0024.html>, Last Access: October 10, 2013.
- Rosenberg, J. et al: RFC3264: An Offer/Answer Model with the Session Description Protocol (SDP), June 2002, <http://www.ietf.org/rfc/rfc3264.txt>, Last Access: October 10, 2013.
- Sesame REST API: <http://openrdf.callimachus.net/sesame/2.7/docs/users.docbook?view>, Last Access: October 10, 2013.
- Skype: <https://support.skype.com/en/faq/FA1417/how-much-bandwidth-does-skype-need>, Last Access: October 10, 2013.
- Troncy, R. et al (ed): Media Fragments URI 1.0 (basic), W3C Recommendation 25 September 2012, <http://www.w3.org/TR/media-frags/>, Last Access: October 10, 2013.
- Tummarollo, G. et al: RDFSyc: efficient remote synchronization of RDF models, The Semantic Web, Lecture Notes in Computer Science, Volume 4825. ISBN 978-3-540-76297-3. Springer-Verlag Berlin Heidelberg, 2007, p. 537, <http://iswc2007.semanticweb.org/papers/533.pdf>, Last Access: October 10, 2013.
- WebRTC Chrome: <http://www.webrtc.org/chrome>, Last Access: October 10, 2013.
- WebRTC Firefox: <http://www.webrtc.org/firefox>, Last Access: October 10, 2013.

Integrating Adaptation and HCI Concepts to Support Usability in User Interfaces

A Rule-based Approach

Luisa Fernanda Barrera, Angela Carrillo-Ramos, Leonardo Florez-Valencia,
Jaime Pavlich-Mariscal and Nadia Alejandra Mejia-Molina
Departamento de Ingeniería de Sistemas, Pontificia Universidad Javeriana, Bogotá, Colombia
{luisa.barrera, angela.carrillo, florez-l, jpavlich, nadia.mejia}@javeriana.edu.co

Keywords: User Interfaces, Adaptation, Usability, HCI.

Abstract: A common problem in information systems development is to provide support for *adaptation*, to automatically adjust their services to different users and contexts. User Interfaces (UI) are required to adapt to those contexts and to satisfy specific criteria and standards to guarantee usability. Several methods have been created to ensure a degree of usability in UI. However, these methods focus mainly in the design stage of the development process. The benefits of these methods may be lost during execution time, since they do not address the necessity to dynamically adapt the interfaces both to context and users. To address this issue it is necessary to integrate User Interface Design with Adaptation, to ensure that UI usability is preserved at the execution time, for different users and contexts. This paper proposes the framework *Tukuchiy*, a rule-based system that dynamically generates Adaptive User Interfaces, based in HCI precepts. This guarantees their usability during execution time, while taking into account user preferences and context. This paper focused in the rule-based system of *Tukuchiy*. That rule system includes usability criteria commonly used for web pages, which were mapped to a desktop application.

1 INTRODUCTION

When users interact through a computational system, they do it through a User Interface (UI). An adequate user interface design has become a very important aspect in software development (Stone et al., 2005). This problem is studied by two areas: HCI and Adaptation. Human-Computer Interaction (HCI) is a discipline that utilizes ideas from Psychology, Ergonomics, and other disciplines, to improve usability of user interfaces and provide better interaction between users and systems. Adaptation considers the heterogeneity of users and the context in which they utilize computers and requires that UI could be easily adapted to perform various tasks.

User interface design in HCI is commonly performed at the design stage in software development, but some usability characteristics defined at the design stage are lost during execution time. Adaptation, however, does not prescribe ways to improve usability of information systems. To address this issue, previous work of the authors proposed *Runa-Kamachiy* (Barrera et al., 2013a), a

model to integrate HCI and Adaptation concepts, to improve the interaction between user and Adaptive systems, and improve usability. To validate the *Runa-Kamachiy* model, a framework, called *Tukuchiy* (Barrera et al., 2013c) was created. *Tukuchiy* realizes *Runa-Kamachiy* as an infrastructure to generate dynamic user interfaces. Two prototypes were created to validate the model in two application areas: *Idukay* (Barrera et al., 2013c) for education and *Midiku* for clinic radiology.

This paper describes a rule-based system utilized by *Tukuchiy* to dynamically adapt user interfaces, and the way Adaptation concepts can be utilized to improve usability of user interfaces.

The remainder of this paper is organized as follows. Section 2 explains basic concepts required to understand *Tukuchiy*. Section 3 reviews related work. Section 4 describes *Tukuchiy* and the way it addresses the Nielsen criteria. Section 5 details *Tukuchiy*'s rule-based system for UI generation and its application in the *Midiku* prototype. Section 6 describes the validation of the prototype. Section 7 concludes and describes future work.

2 BACKGROUND

This section describes background concepts required to understand Tukuchiy.

First HCI concept that we use is the “Five User Interface Laws”. Hale *et al.* (Hale, 2011) indicate that there are five laws that every user interface designer should know and apply: *i) Fitts Law* (Guiard and Beaudouin-Lafon, 2004); *ii) Miller Law* (Miller, 1956); *iii) Steering Law* (Accot and Zhai, 2001); *iv) Hicks Law* (Seow, 2005); *v) Practice Law*, (Roessingh and Hilburn, 2000).

The second HCI concept is “Nielsen's Usability Heuristics. Nielsen proposes ten heuristics to design user interfaces (Nielsen, 1994). They are as follows: *i) System state visibility; ii) Coincide real world and system; iii) User control and freedom; iv) Consistency and Standards; v) Error prevention; vi) Recognizing instead of remembering; vii) Flexibility and efficiency; viii) Static and minimalistic design; ix) Help the user to recognize, diagnose and recover from errors; x) Help and Documentation*, documentation should be easy to search, focus in user task.

In addition, Tukuchiy utilizes two main Adaptation concepts: User, and Context Profiles. User profiles represent tastes, necessities, and preferences of each user in a system, and can be used to adjust the services provided by the system, according to individual user aspects. Context profiles represent the user environment, characteristics that may affect the system's usability. Particularly, Tukuchiy takes into account the time of the day to adjust UI illumination. This adjustment is based in the Berry criteria (Berry, 2013), which indicates the way to manage brightness to ensure that user interface colors are comfortable for the user and would not reduce his/her perception capabilities.

3 RELATED WORK

Related Work Table 1 shows a comparison between related work about UI usability and UI generation, based in our work in (Barrera *et al.*, 2013c) and (L. F. Barrera *et al.*, 2013a). Columns are the related works. Rows are the criteria to evaluate each work. The columns 1-6 are as follows: 1(Moussa *et al.*, 2000); 2 (Criado *et al.*, 2010); 3 (Zimmermann *et al.*, 2013); 4 (Namgoong *et al.*, 2006); 5 (Akoumianakis and Stephanidis, 1997); 6 (England *et al.*, 2009).

The criteria used in Table 1 were chosen to highlight the deficiencies with respect to interfaces

usability. The evaluation comprises both HCI and Adaptation. Most works do not focus on improving usability. Although most take into account user profile and his/her context, they do not take into account HCI standards. These works do not take into account that interfaces change during execution time and that it is necessary to avoid losing standards given during design time.

Table 1: Related Work Comparison.

| Criterion | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|-----|---|---|---|---|
| Takes into account usability criteria during Execution (E) time or Design (D) time. | D | E-D | D | D | D | E |
| UI let the user recognize, diagnose, and recover from errors | - | - | + | - | - | + |
| UI include help and documentation | + | - | - | - | + | - |
| Keeps consistency between the real world and the system | + | + | - | + | + | - |
| Adapts to different types of users | - | + | - | - | - | + |
| Takes into account user context aspects | - | + | + | + | - | - |
| Uses HCI techniques | + | - | - | - | + | + |
| Utilizes a rule-based system to generate UI | - | - | - | - | + | - |

4 TUKUCHIY

Tukuchiy ("*Tukuchiy*" is a Quechua word that means "*To Transform*") is a framework based on the Runa-Kamachiy model (L. F. Barrera *et al.*, 2013a), to generate dynamic user interfaces, adjusted to specific user characteristics, context, and presentation preferences.

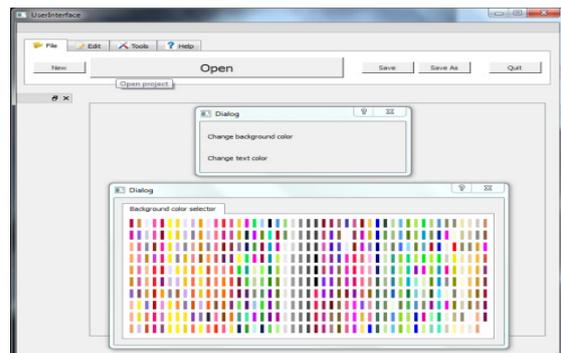


Figure 1: Tukuchiy Base Interface.

Tukuchiy keeps some usability standards at execution time, so that UI usability can be kept across the entire life cycle of the system (Figure 1). The Figure 1 shows some of the usability rules in Tukuchiy.

The description of Tukuchiy's component and

Table 2: Usability Criteria and Heuristics.

| Usability Criterion | Heuristic |
|-------------------------|--|
| Learning | User control and freedom. Recognize instead of Remembering. |
| Error prevention | Help the user to recognize, diagnose, and recover from errors. Help and Documentation. |
| Memorization | Consistency and Standards. Help and Documentation. |
| Efficiency | Flexibility and Efficiency. |
| Satisfaction | System state visibility. |
| Efficacy | Static and minimalistic design. |

conceptual integration can be found in (Barrera et al., 2013b). This section focuses in explaining the way Tukuchiy adapts usability concepts in web pages to desktop applications. Table 2 shows the usability criteria addressed by this research and the heuristics utilized to address them.

4.1 Learning

Learning is related to the capacity of the software to let users learn to use its components (Carvajal and Saab, 2010). Tukuchiy uses Practice Law (Section 2.1) to provide different types of help, with various levels of detail, depending on the user expertise. If the user is new, help is more detailed (see Figure 2a). As the user gains more experience utilizing the system, help is reduced (Figure 2b).

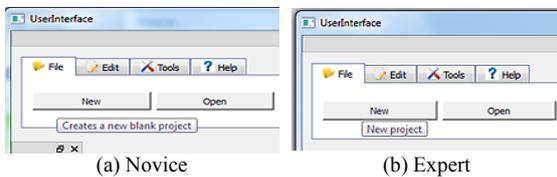


Figure 2: Learning in Tukuchiy.

4.2 Error Prevention

An error-tolerant interface is designed to assist the user in recovering from errors (Carvajal and Saab, 2010). Tukuchiy utilizes tooltips with more or less information, according to the user experience level. Users with less experience received more detailed tooltips, while users with more experience received tooltips with less detail.

Additionally interface buttons are associated to intentionality and a color that represents that intentionality. Figure 12 shows the color palette utilized when one wants to change the color to the "close" button. Since this button is associated to a "danger" intentionality, blue and green colors, which represent "harmony", are not present in the palette.

4.3 Memorization

The memorization aspect enables users to easily remember how to interact with that system, after a period without using it (Nielsen, 1995). To address the memorization aspect, this research utilized the Miller Law (section 2.1). Tukuchiy groups buttons according to functionality based in this Miller rule (see Figure 3). Changes performed by the user in design (personalization) of the UI persist across sessions.

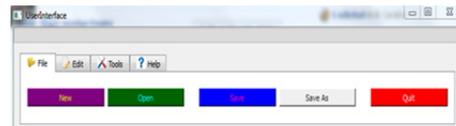


Figure 3: Memorization in Tukuchiy.

4.4 Efficiency

Efficiency is associated to the amount of effort required by the user to achieve a specific goal in his/her interactions with the system (Carvajal and Saab, 2010). Tukuchiy enlarges buttons (see Figure 8), based in the Fitts Law. In addition, it reduces in the amount of colors in palettes. These two strategies reduce the user efforts to accomplish a task or to personalize the interface.

4.5 Efficacy

According to ISO 9131-11, efficacy is the degree in which planned activities are performed and the planned results are achieved. In other words, can users do what they need in a precise manner?

The use of tooltips and Fitts Law to enlarge UI elements, seeks to improve the precision of the performed tasks. In addition, color transformation assists people with color blindness to properly identify colors and avoid mistakes. As seen in Figure 6, for people with Protanopy (red color blindness), Tukuchiy changes the color palette, discarding red colors, so that the user may distinguish a broader range of colors.

4.6 Satisfaction

Satisfaction is the perception of pleasantness and positive attitude towards the utilization of a product. That perception is reflected in the physical and emotional actions of the user when utilizing the system (Carvajal and Saab, 2010). The system does not directly address this criterion. However, we sought to indirectly satisfy the user by integrating all

of the other criteria. For instance, the palette change for color blind users (Flück, 2006) may be pleasant for them.

5 RULE SYSTEM

To maintain usability characteristics during execution time, Tukuchiy utilizes a rule-based system, which is detailed in this section. To test the rules, *Midiku* was built, a clinic radiology application, to support the diagnostic process and medical image simulation. To build the system, two groups of rules were created: HCI rules and Adaptation rules. Both are detailed in the following sections.

5.1 HCI Rules

This group of rules realizes a subset of HCI standards. To build that subset, this research verified all of the standards that could be kept during execution time. The rules are the following:

5.1.1 Physical Conditions

The system focuses in assisting two physical difficulties: color blindness and myopia. Two processes are performed to assist in these difficulties: color simulation and polarization and button enlargement.

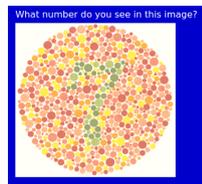


Figure 4: Ishihara Test (Flück, 2009).

For color blindness, Tukuchiy changes the palette using the following rule: *i*) Color blindness identification: when entering the system, the user is shown an image corresponding to the Ishihara Test (Flück, 2009). This test determines which type of color blindness the user (see Figure 4); *ii*) based in the code of (Duck, 2012), a simulation is performed in which palette colors are changed, so that they could be perceived by the person, according to his/her color blindness type.; *ii*) base colors are compared with simulated colors and the difference is calculated. This is used to change colors that are visible to the user.

Figure 5 is a fragment of the rule for color

changes. This rule is utilized when the user needs to use the palette to personalize the interface colors.

```

{ // Initial LMS
  float l, m, s;
  float L = (17.8824f * r) + (43.5161f * g) + (4.11935f * b);
  float M = (3.45565f * r) + (27.1554f * g) + (3.86714f * b);
  float S = (0.0299566f * r) + (0.184309f * g) + (1.46709f * b);
  ...
  else if (enfermedad=="Tritanope"){
    l = 1.0f * L + 0.0f * M + 0.0f * S;
    m = 0.0f * L + 1.0f * M + 0.0f * S;
    s = -0.395913f * L + 0.801109f * M + 0.0f * S;
  }
  ...
}
    
```

Figure 5: Color transformation rule fragment (color blindness).

The Ishihara test is performed several times, to mitigate any external factors that could affect the validity of the user answers (e.g. screen resolution).

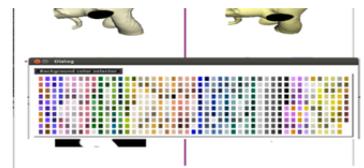


Figure 6: Midiku Color Transformation, user with protanopia (Barrera et al., 2013).

As shown in Figure 6, this rule changes colors both to the buttons and the medical image being examined.

The following algorithm is used for button enlargement: *i*) Identify the visual problem is explicitly, asking the user if he/he has myopia, this is stored in the user profile; *iii*) button properties are changed, so that, whenever the user points to the button, it changes its size. The layout of buttons within the same functional group is re-arranged.

```

void UserInterface::ButtonMouseOver_ui() {
  focus=focusWidget();
  if (focus->inherits("Button"))
  {bool fittsTrue=rules->evaluateFitts(user->
  userPhysical.getScaleFuntionalDiversity());
  if (fittsTrue) {(Button*) focus->changeSizeFitts();}}
}
    
```

Figure 7: Button enlargement rule.

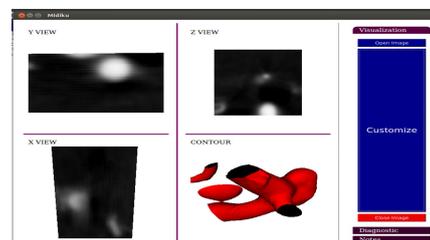


Figure 8: Button Enlargement.

Figure 7 describes the way button scale is changed according to the user profile. This rule is based in

the Fitts Rule (Guiard and Beaudouin-Lafon, 2004). Figure 8 shows an example of button enlargement in Midiku.

5.1.2 Effective Color Combinations

Wright et al (Wright et al., 1997) indicates that colors are not visible when they overlap. To ensure that this rule is enforced during execution, Tukuchiy performs the following algorithm: *i*) identify the color of the element that contains the component whose color is going to change; *ii*) identify the color of the elements contained by the component whose color is going to change; *iii*) using the above information, the color palette is filtered to eliminate the colors that, according to Wright, do not match adequately (Wright et al., 1997). Color combinations are organized in a pessimistic manner, i.e., there is a list of colors that do not match in the previous steps; *iv*) Presentation of the color palette, the filtered palette is presented to the user.

```
void Rule::changeCustomPalette(Button *w) {
    removeOriginalStyleColorGroup(w);
    removeParentColorGroup(w);
    removeChildrenColorGroups(w);
    removeIntentionGroupColor(w);
}
```

Figure 9: Color combination rule.

Figure 9 shows the palette change rule according to color combination.



Figure 10: Color combination example.

Figure 10 is an example of a palette change, in which the text of the red button contains only the colors that combine or contrast with the button color.

5.1.3 Widget Intentionality

In a study performed by Bedolla (Bedolla, 2002), colors are associated to specific psychological states and have specific intentionality (e.g. red is associated to danger situations). This is taken into account to assign to each UI element, an intentionality associated to a color, to keep each element's essence. The algorithm for this task is the following: *i*) a table is created that maps intentionality to allowed and forbidden colors, an XML file is used to store that table. *ii*) when the user is going to change a color, it is evaluated whether

the color keeps the same intentionality, according to the table. Similarly to the color combination rule, this filter is performed pessimistically and is presented as a palette to the user

```
void Rule::removeIntentionGroupColor(QWidget *w) {
    Dialog * auxDialog=((Button*)w)->getMenu()->getDialog();
    evaluateColorButtonIntention(w->objectName().toString());
    vector<Intention> vecAux = intentions->getIntentions();
    vector<string> colorIntention;
    for(int d=0;d<vecAux.size();d++)
    {for(int y=0;y<this->buttonIntention.size();y++)
        {if(vecAux[d].getIntentionName()==this->buttonIntention[y]){
            colorIntention = vecAux[d].getForbiddenColors();
            for(int j=0;j<colorIntention.size();j++){
                ((Button *)w)->getMenu()->removeColorFromPalettes(colorIntention[j]);
            }
        }
    }
    ((Button*)w)->getMenu()->setDialog(auxDialog);
}
```

Figure 11: Button intentionality.

Figure 11 is a fragment of the rule that performs the filter (eliminate colors of the palette) of the colors that are not allowed, according to the button intentionality. Figure 12 shows the allowed colors for the "close" button.

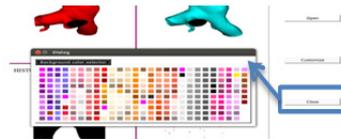


Figure 12: Widget Intentionality in Midiku.

5.1.4 Luminosity

Given the importance of luminosity (see Section 2.3), a rule was created to transform the colors of the entire interface. The transformation takes context into account. For instance, illumination is changed according to the time of the day.

```
RGB colorblind::brightness(RGB color, int delta){
    this->R=color.red+delta; this->G=color.green+delta;
    this->B=color.blue+delta;
    this->R = qBound(0, (int) R, 255);
    this->G = qBound(0, (int) G, 255);
    this->B = qBound(0, (int) B, 255);
    RGB rgbFinal;
    rgbFinal.red=this->R; rgbFinal.green=this->G;
    rgbFinal.blue=this->B;
    return rgbFinal;}
}
```

Figure 13: Color brightness.



Figure 14: Brightness palettes in Tukuchiy.

Figure 13 and Figure 14 is an example of the way illumination is updated according to context. This rule does not perform changes while the system is being utilized, to avoid being too intrusive for the user.

5.2 Adaptation Rules

This section shows the Adaptation rules that complement the dynamic generation of interfaces, based in user characteristics and context.

5.2.1 Help

Tukuchiy filters information during the system startup to evaluate the user experience level and language preferences. This rule process is as follows: *i*) the user profile has an attribute that indicates the amount of time the user utilizes the system, which determines the experience level of the user; *ii*) when starting the system, the user chooses his/her language of preference; *iii*) from the information in *i*) and *ii*), the system changes the names in its UI elements according to the chosen language. Tooltips are automatically changed according to the experience level. Currently, the system has two tooltips that are more detailed for novice users than for expert users.

```

void UserInterface::reloadUserInterface(){
    if(user->useLevel=="Novato"){
        if(this->language=="English"){
            loadUserLevelLaguageFile(":files/messages-en-rookie.xml");
        }else{
            loadUserLevelLaguageFile(":files/messages-es-rookie.xml");
        }
    }else if(user->useLevel=="Experto"){
        if(this->language=="English"){
            loadUserLevelLaguageFile(":files/messages-en-expert.xml");
        }else{
            loadUserLevelLaguageFile(":files/messages-es-expert.xml");
        }
    }
}
    
```

Figure 15: Use level about use level.

Figure 15 illustrates the help rule. Each help has an XML file that associates UI elements with different tooltips and names.

5.2.2 Color Preferences

Each user may have different preferences about colors to display each UI element. To realize this in the preferences, a rule was created that organizes the color palette, to show the UI according to the tastes of the user.

This rule is utilized as follows: *i*) each time a user selects a color, a counter is updated, which is

used by the system to find out which the degree of color preference; *ii*) based in the degree of color preference, the color palette is reorganized from most preferred colors to least preferred colors.

```

void Dialog::initializeBg(map <string,vector<RGB> >
    palette,vector<pair<string,int> > colorPreferences){
    ...
    for(int i=0;i<colorPreferences.size();i++){
        string nombreLlave=colorPreferences[i].first;
        vector<RGB> colores=palette[nombreLlave];
        for(int j=0;j<colores.size();j++,cont++){
            RGB color=colores[j];
            colorblind aux;
            //brightness change
            color = aux.brightness(color, this->delta);
            //disease change
            string cadenaTransformada = aux.corregirColor(color.red,
                color.green, color.blue, this->getPaletteType());
            ...
        }
    }
}
    
```

Figure 16: Color preferences rule.

Figure 16 is a code fragment that denotes the way the palette is organized according to the user preferences. Figure 17 shows the palette that result from applying the above rule. In this example, the preferred color is green.



Figure 17: Color preferences in Midiku.

6 PILOT TEST

The authors are currently developing a functional prototype called Midiku. Since this is a work in progress, its initial assessment has only been performed over the design of Midiku's User Interface. This design includes functionality given by Tukuchiy (see Section 4).

6.1 Evaluation Process

To evaluate the usability of Midiku's interfaces, Mock-ups (Soegaard, 2004) were utilized. Mockups are a digital demonstration of the way the UI will look like in the final system. Mockups were shown to three physicians of the San Ignacio Hospital in Bogotá, Colombia. One of them is an expert radiologist with several years of experience, who is not proficient with current computing technologies. The other two are physicians who are specializing in radiology and have high proficiency utilizing current computing technologies. The three physicians

answered a survey based in QUIS (Questionnaire for User Interface Satisfaction) (Chin et al., 1988). The questions answered focused exclusively in evaluating the UI design.

Table 3: Example questions of the survey.

| Criterion | Question | Scale |
|-----------------------------------|---|--|
| 1. Interaction and Adaptability | Flexibility of the user interface | Very rigid, Rigid, Flexible or Very Flexible |
| | Complexity of the user interface | Very hard, Hard, Easy or Very easy |
| 2. Screen and Display | Organization of information on screen | Very confused, Confused, Clear or Very clear |
| | Is the screen density: | Very inadequate, Inadequate, Adequate or Very adequate |
| 3. Presentation and Visualization | Are groups of info demarcated? | Very confused, Confused Clear or Very clear |
| | Does it provide visually distinctive data fields? | Very high grade, High grade, Low grade or Very low grade |

Table 3 shows some of the questions, grouped by evaluation criteria. The first criterion is the user appreciation with respect to the interface. The second criterion is the organization and meaning of graphical elements in the screen. The third criterion is about the color utilization and screen zones delimitation.

6.2 Pilot Test Results

Surveyed subjects were divided in two groups: expert and novice. Three questions from Criterion 1 (Interaction and Adaptability), six from Criterion 2 (Screen and Display), and four from Criterion 3 (Presentation and Visualization). Figure 18 indicates the results of the survey.

For each criterion, results are shown for the expert group, the novices group and the expected value, which is the maximum score that can be obtained in each criterion.

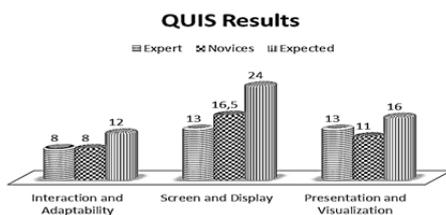


Figure 18: Survey Results.

The Figure 25 indicates that the expert radiologist valued the first criterion as 66.7%, emphasizing the interface flexibility, but he expressed that the attractiveness of interface has a low level. He valued the second criterion as 54.2%, emphasizing the adequate density of elements in the screen, but indicating the difficulty to understand the meaning of buttons. He valued the third aspect as 81.3%,

emphasizing the adequate use of colors.

For novice users, the answers were averaged. The first aspect was valued as 66.7%, emphasizing the ease of initial interpretation of the interface. The second aspect was valued as 68.8%, emphasizing the organization and adequate terminology, but they expressed the density of elements in the screen are inadequate. The third aspect was valued as 68.8%, emphasizing the adequate visual distinction among screen zones, but indicating the inadequate utilization of colors.

The users commented that they would want to have more intuitive and less complex radiology interfaces. They also commented that there are “dead spaces” in the screen that could be better utilized to present information. They indicated that the survey could be enriched by using videos of the mockups, to better understand the functionality.

7 CONCLUSIONS AND FUTURE WORK

This paper presented Tukuchiy, a framework that integrates several methods and techniques in HCI with Adaptation concepts to improve user interaction with systems in changing contexts. Tukuchiy's rule-based system ensures usability criteria are preserved at execution time in changing interfaces.

This paper also presented a functional prototype (Midiku) that supports radiologists to diagnose medical images. An initial assessment at this stage has only been performed over the UI design of Midiku. This assessment was performed through a Mockup and a survey that was answered by an expert and two novice radiologists. The results emphasize positive aspects, such low UI complexity, adequate organization of information on the screen and the ease to visually distinguish data fields. Negative aspects found are the difficulty to understand the meaning of buttons, inadequate characters visualization and inadequate terminology.

Future work includes fully developing the functionality of Midiku and performing a more detailed analysis of the capabilities of Tukuchiy in terms of efficiency, efficacy, and user satisfaction. In addition, Tukuchiy will be assessment with fully test (20 observer's approx.). The assessment includes an initial perception test and then interaction atoms test (key functionalities).

ACKNOWLEDGEMENTS

The authors want to acknowledge the Systems Engineering Department and the Systems Engineering and Computing Master's program of the Pontificia Universidad Javeriana, for supporting the development of *Runa-Kamachiy*, Tukuchiy and Midiku. The authors also thank Luis Felipe Uriza and his team of resident physicians of the San Ignacio Hospital, for their collaboration in the initial evaluation of Midiku.

REFERENCES

- Accot, J., Zhai, S., 2001. Scale effects in steering law tasks, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01. ACM, New York, NY, USA, pp. 1–8.
- Akoumianakis, D., Stephanidis, C., 1997. Supporting user-adapted interface design: The USE-IT system. *Interacting with Computers*, No 3, Vol 9, pp. 73–104.
- Barrera, L., Mejía-Molina, N., Carrillo-Ramos, A., Flórez-Valencia, L., 2013. Tukuchiy - Color Palettes. URL <http://s24.postimg.org/xf3eqh4px/paletas.png>. (8.20.13).
- Barrera, L. F., Carrillo-Ramos, A., Flórez-Valencia, L., Pavlich-Mariscal, J. A., 2013a. Runa-Kamachiy: Conceptual Integration Model Between HCI and Adaptation Oriented to User Interface Usability. *DYNA -Rev. Fac. Minas Univ. Nac. Colomb. Sede Medellín En Revisión*, 10.
- Barrera, L. F., Mejía-Molina, N. A., Carrillo-Ramos, A., Flórez-Valencia, L., Pavlich-Mariscal, J., 2013b. . UMUAI - USER Model. *USER-Adapt. Interact. J. Pers. Res. En Revisión*, 40.
- Barrera, L. F., Mejía-Molina, N.A., Carrillo-Ramos, A.C., Flórez-Valencia, L., 2013c. Tukuchiy-Idukay: Generación Dinámica de Interfaces en Contextos Educativos. *Presented at the Congreso Internacional de Ambientes Virtuales de Aprendizaje Adaptativos y Accesibles*, San Juan, Argentina, p. 10.
- Bedolla, D., 2002. Diseño sensorial, las nuevas pautas para la innovación, especialización y personalización del producto. *Universitat Politècnica de Catalunya. Dept de Projectes d'Enginyeria*, Barcelona, España.
- Berry, C., 2013. Guide to procedures of the N.C. *Occupational Safety and Health Review Commission*, 2nd ed. Raleigh, North Carolina, USA.
- Carvajal, M., Saab, J., 2010. Fundamentos conceptuales de las Directrices de Usabilidad de Gobierno en línea., Gobierno en Línea. *Ministerio de Tecnologías de la Información y Comunicaciones*, Bogotá, Colombia.
- Chin, J. P., Diehl, V.A., Norman, K. L., 1988. Development of an instrument measuring user satisfaction of the human-computer interface, in: *Proceedings of the Conference on Human Factors in Computing Systems*, CHI '88. ACM, New York, USA, pp. 213–218.
- Criado, J., Vicente-chicote, C., Padilla, N., Iribarne, L., 2010. A Model-Driven Approach to Graphical User Interface Runtime Adaptation. Presented at the 5th Workshop on Models@run.time at the ACM/IEEE *13th International Conference on Model Driven Engineering Languages and Systems (MODELS 2010)*, Oslo, Norway, pp. 49–59.
- Duck, D., 2012. DALTONISM: Making Games Color Blind Friendly SebbyLive - My Life My Proj. URL <http://www.sebbylive.com/2011/08/03/daltonism-making-games-color-blind-friendly>.
- England, D., Randles, M., Taleb-Bendiab, A., 2009. Runtime user interface design and adaptation, in: *Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology*, BCS-HCI '09. British Computer Society, Swinton, UK, UK, pp. 463–470.
- Flück, D., 2006. Color Blind Essentials. Colblindor, Zürich, Switzerland.
- Flück, D., 2009. Ishihara Color Test. Color Blind. URL <http://www.colour-blindness.com/en/colour-blindness-tests/ishihara-colour-test-plates/>
- Guiard, Y., Beaudouin-Lafon, M., 2004. Fitts law 50 years later: applications and contributions from human-computer interaction. *International Journal of Human-Computer Studies*, No 6, Vol 61, pp. 747–750.
- Hale, K., 2011. 5 Interface Laws Every Software Designer Should Know (WWW Document). URL <https://speakerdeck.com/roundedbygravity/5-interface-laws-every-software-designer-should-know> (accessed 7.25.13).
- Miller, G.A., 1956. The magical number seven, plus or minus two: *some limits on our capacity for processing info*. *Psychological Review*, No 2, Vol 63, pp. 81–97.
- Moussa, F., Kolski, C., Riahi, M., 2000. A model based approach to semi-automated user interface generation for process control interactive applications. *Interacting with Computers*, No 3, Vol 12, pp. 245–279.
- Namgoong, H., Sohn, J.-C., Cho, Y.-J., Chung, Y.K., 2006. An Adaptive User Interface in Smart Environment exploiting Semantic Descriptions, in: *2006 IEEE Tenth International Symposium on Consumer Electronics*, 2006. ISCE '06. Presented at the 2006 IEEE Tenth International Symposium on Consumer Electronics, 2006. ISCE '06, IEEE, pp. 1–6.
- Nielsen, J., 1994. Enhancing the explanatory power of usability heuristics, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94. ACM, New York, NY, USA, pp. 152–158.
- Nielsen, J., 1995. Usability 101: Introduction to Usability (WWW Document). Jakob Nielsen Website. URL <http://www.useit.com/alertbox/20030825.html> (5.14.12).
- Roessingh, J. J. M., Hilburn, B. G., 2000. The Power Law of Practice in Adaptive Training Applications, *1st ed. Nationaal Lucht- en Ruimtevaartlaboratorium*, Amsterdam, Netherlands.
- Seow, S. C., 2005. Information theoretic models of HCI: a comparison of the Hick-Hyman law and Fitts' law. *Human Computing Interaction*, #3, V20, pp. 315–352.

- Soegaard, M., 2004. Mock-ups (WWW Document). Interact. Des. Found. URL www.interaction-design.org/encyclopedia/mock-ups.html (accessed 10.15.13).
- Stone, D., Jarrett, C., Woodroffe, M., Minocha, S., 2005. User Interface Design and Evaluation, *1st Edition*. ed. *Morgan Kaufmann Series in Interactive Technologies*, San Francisco, USA.
- Wright, P., Mosser-Wooley, D., Wooley, B., 1997. Techniques and Tools for using color in computer interface design. *Crossroads ACM Student Magazine*, No 3, Vol 3, pp. 3–6.
- Zimmermann, G., Jordan, J. B., Thakur, P., Gohil, Y., 2013. GenURC: generation platform for personal and context-driven user interfaces, in: *Proce. of the 10th International Cross-Disciplinary Conference on Web Accessibility, W4A '13*, New York, USA, pp. 6:1–6:4.

***Tactive*, a Framework for Cross Platform Development of Tabletop Applications**

Ombretta Gaggi¹ and Marco Regazzo²

¹*Department of Mathematics, University of Padua, via Trieste, 63, Padova, Italy*

²*Anyt1me S.r.L., via Siemens 19, Bolzano, Italy*
gaggi@math.unipd.it, marco.regazzo@anyt1me.com

Keywords: Tabletop Applications, Touch Interfaces, Webkit Engine.

Abstract: The number and types of applications developed for multi-touch tabletops are dramatically increased in the last years, mainly due to the fact that interactive tabletops allow a more natural interaction with the user through their multi-touch interfaces. Despite many applications share a big set of common features, e.g., gestures recognition, interface orientation, etc., almost all applications implement their home made software solutions. In this paper we present *Tactive*, a software layer for fast development of *portable* applications for multi-touch interactive tabletops. *Tactive* allows to abstract from hardware and software equipment and to embed a web application into a application for multi-touch surfaces. Our framework supports up to five fingers gestures recognition and communication between different windows, and allows to save more than 60% of developing time.

1 INTRODUCTION

In the last few years, the market of multi-touch tables is experiencing a situation very similar to what happens in the mobile applications market. The number of developed applications is dramatically increased, interactive tabletop surfaces are used to improve learning activities (Rick et al., 2011), inside museums (Geller, 2006), where the diversity of visitors create a natural laboratory for testing this kind of interface, to help the management of emergency (Qin et al., 2012), and in many other collaborative activities like, e. g., photoware (Pedrosa et al., 2013), etc. Consequently, also the the number of hardware solutions increased, each one requiring a particular SDK, programming language, etc.

Like smartphones, interactive tabletops allow a more natural interaction with the user through their multi-touch interfaces (Forlines et al., 2007) and, unlike mobile devices, they allow the interaction of more than one user at the same time. Tabletops promote collaboration and social experiences, and can act as a meeting point.

The possibility to interact with multiple user at the same time requires an important set of new features for the user interface: e. g. the user are placed around the table, therefore they need different orientations of the interaction widgets, they can collaborate

using the same space and objects or can compete for them. Therefore all applications developed for this kind of interface have to face a common set of problems, e. g., the recognition and management of particular gestures and the orientation of the interactive widgets. Despite these common features, almost all the applications implement all these features starting from scratch, since no software solution exists.

In this paper we present our tabletops solution, which can be applied to different size of surface (42" or more) and can work upon different operating system. We have developed a software layer which is able to abstract from hardware and software constraints of the device in which it is installed (screen size, operating system, etc) and allows the developer to easily manage common features of the user interface discussed above. Our solution provide a Javascript API which allows a developer to build an entire tabletop application only using web technologies, in particular HTML5, CSS3 and Javascript, without the need to know anything about the underlying hardware and software.

Moreover, our solution allows a very easy reuse of web applications already developed for non touch interfaces, and personalization of the final application.

Finally, we provide our multi-touch tabletop with a strong interior design (see Figure 1) on its shape and materials while most of the hardware available on the



Figure 1: Our tabletop solution was designed by an expert interior architect.

market is little more than a “big iPad mounted on four legs”.

The paper is organized as follows: Section 2 discusses the related works and the need for a framework for the development of portable multi-touch applications. Section 3 presents *Tactive*, a software layer which provides a set of features and gestures to speed up the design process of multi-touch interactive applications. A set of success stories about applications developed with the framework is discussed in Section 4. Finally, we conclude in Section 5.

2 RELATED WORKS AND BACKGROUND

There are a lot of applications for interactive tabletops and surfaces described in literature. Correia et al (Correia et al., 2010) described an application for museum setting. A tabletop is used to enhance user experience presenting semantic information about all the artworks of the exhibition. The authors realized a tabletop setup based on the Frustrated Total Internal Reflection system. More than one user can interact with the tabletop at the same time in a collaborative way. The user interface is an ad-hoc application, build from scratch with the help of some open framework.

uEmergency (Qin et al., 2012) is a emergency management system based on a very large multi-touch surface. The users can collaborate in a very intuitive way around a table which displays available information on the ongoing emergency. It allows people to carry out face-to-face communication based on a horizontal map. Users can also analyzed real-time situation with fingers or digital pens. A study shows that the use of interactive surfaces improves efficiency in decision making and collaboration for coping with an emergency.

Pedrosa et al, used tabletops to explore home videos and photos (Pedrosa et al., 2013). A set of 24 users evaluates an application which displays photos

and videos on a horizontal touch surface to allow storytelling and random exploration. The authors show that, among collaborative tools also personal spaces within the tabletop were useful for allowing independent navigation.

The application described so far have many common features: they are highly visual systems, mainly controlled by touches and some common gestures performed on the surface of the system, e. g., browsing a collection of items, selecting a particular item, accessing a documents, and so on. All these applications can interact with several users at the same time, and each user requires a different orientation of the interface, according to his/her position. Despite many common requirements, the developers of all these applications need to implement the majority of these features from scratch and the available frameworks provide only very low level features.

As a general remark, designer of multi-touch applications do not have a reference model to model user interface and interaction, but often rely on best practice and intuition rather than on a systematic development process (Wigdor et al., 2009). For this reason, many works in literature address the problem of designing user interface for interactive surfaces (Anthony et al., 2012; Hesselmann et al., 2011; Luyten et al., 2010; Nielsen et al., 2004; Seto et al., 2012; Urakami, 2012).

Urakami (Urakami, 2012) has shown that the user choice of gestures was affected by the size of the manipulated object, expertise, and nature of the command (direct manipulation of objects vs. assessment of abstract functions), therefore it is essential to involve the user in the development of gesture vocabularies. The same approach is followed by Hesselmann et al, that proposed an iterative process of five steps tailored to the development of interactive tabletops and surfaces applications, called SCiVA, Surface Computing for Interactive Visual Applications. The key idea of SCiVA is to strongly involve the user in the design process to improve the usability of the final product (Hesselmann et al., 2011).

Luyten et al, try to reach a consensus on a set of design patterns that aid in the engineering of multi-touch interfaces and transcend the differences in available platforms (Luyten et al., 2010). Seto et al, investigate the problem of how to manage menus displacement in multi-user surfaces (Seto et al., 2012). In particular they focus on the discoverability of system menus on digital tabletops designed for public settings. This study presents a set of design recommendations to improve menu accessibility: e. g., discernible and recognizable interface elements, such as buttons, supported by the use of animation, can effec-

tively attract and guide the discovery of menus.

This analysis of the literature shows that some steps toward the definition of design patterns for the development of interactive multi-touch interfaces have been done, but there are not already built *off-the-shelf* components to create these interfaces, but each application build from scratch its user interface. Native frameworks, like Microsoft Surface 2.0 SDK and Runtime (Microsoft, 2013a), Windows Presentation Foundation + Native Touch recognition by Microsoft Windows 8 (Microsoft, 2013b) and Smart Table SDK (SMART Technologies, 2013), help to develop multi-touch applications but require a particular hardware/software configuration.

Our goal is the creation of a software layer, portable on each operating system and hardware solution, which provides this set of features and gestures to speed up the design process of multi-touch interactive applications by avoiding to re-invent the wheel each time (Gaggi and Regazzo, 2013).

Other solutions exist which addresses a similar problem. Glassomium (Toffanin, 2013) is a project based on web technologies which aims to port web applications to multi-touch surface. Even if the key idea is quite the same, it allows for rotations, scaling and dragging even through an unstable beta and it is not able to identify gestures which involve the whole hand, Glassomium can be considered a windows manager, which allows to recognize the user gestures and to manage them, but it does not implements cross-windows communication, therefore, it lacks of a proper mechanism to change the user experience on the base of the interaction of other users. To the best of our knowledge this feature is implemented only by our solution.

GestureWorks (Ideum, 2013) and Arena (Unedged, 2013) are frameworks which provide generic and cross platform functionalities, like gestures recognition, to develop touch applications, but they are not able to manage more than one application being launched at the same time or multiple application enclosed in different windows.

3 DESCRIPTION OF THE FRAMEWORK

In this section we discuss the design issue and the implementation details of the developed software layer, called *Tactive*. *Tactive* is a framework, which allows to speed up the development of applications for multi-touch surfaces. This goal is reached since:

- *Tactive* provides a way to encapsulate web applications into widgets suitable for multi-touch sur-

faces, therefore already developed web applications can be easily adapted to multi-touch interactive surfaces;

- *Tactive* allows to abstract from hw/sw details: an entire application can be developed using web technologies, therefore we do not ask the developer to know any particular language or technology bound to the particular hw/sw equipment, he or she only needs to know how to use the Javascript API provided by our framework;
- applications developed with *Tactive* are able to adapt themselves to different size of the surface (*Tactive* helps to realize the so-called *fluid* applications) and
- *Tactive* provides a set of features common to multi-touch applications like windows disposition, gestures recognition and interface orientation.

3.1 System Architecture

The architecture of our system is depicted in Figure 2. *Tactive* is organized in two levels. The lower one, called the *O.S. Layer* guarantees the independence from the underlying hardware: it contains the operating system (MS Windows 7, MS Windows 8 and Linux are supported), and a set of protocol and libraries to manage touch gestures if the chosen operating system does not support them natively. The *Application Layer* manages the applications, their windows and the interaction between the applications and the user or between different applications.

Tactive clearly separates the contents, displayed to the users, from the interaction widgets and the software components used to display the contents. For this reason, the architecture of our framework contains a content manager, called *Application Container*, which manages how to display the contents, and a windows manager.

The Application Container allows the division between contents and interaction widget using *WebViews*, i. e., components that display web pages. A *WebView* allows to embed HTML pages inside an application. This component uses the *WebKit* rendering engine to display web pages inside a window of the application, and includes methods to navigate forward and backward through a history, to zoom in and out, to perform text searches, etc.

The Application Container is the underlying component that encapsulate all the functionalities needed to interact with the user and with other components within the table, i. e., the *Touch Manager* that allows gestures management and recognition, and the

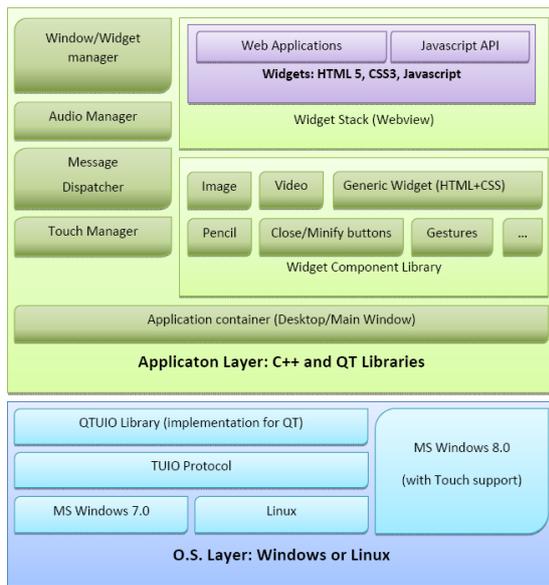


Figure 2: Architecture of the developed software layer for *Tactive*.

Window/Widget Manager that provides the stack of visible objects (see Figure 2). It is also responsible to collect and enumerate application specific contents (e. g., images, videos, web pages or multimedia items) that are stored as web pages and rendered through the *WebView*.

The framework supports both *on-line* and *off-line* content/pages but usually the second option (a local web server) is preferred to let the application works and displays contents even in absence of an Internet connection.

Widgets for the visualization of media items like videos and images have been implemented using *WebViews*. *Tactive* has been designed to be extendible: an expert developer may create a new component extending the widget component (or one of its subclasses), automatically taking advantage of all the features already implemented and described above¹.

Using our framework, content can be created by a web developer (that designs the structure) and update by a content editor.

The mechanism of *WebViews* is used to developed hybrid applications for mobile devices, i. e., applications based on the HTML5 languages which are wrapped with a webkit engine and rendered as native mobile applications. PhoneGap (Apache Software Foundation, 2013), also known as Apache Cordova, is a framework for cross-platform mobile de-

¹We must note here that the framework development is almost complete, therefore, even if *Tactive* is extendible, it is very difficult that a developer of applications needs to implement a new type of widget.

velopment which create hybrid applications. Our approach is very similar: the idea is to take advantage of the portability of web technologies to develop portability of applications for multi-touch interactive surfaces.

Using a *WebView*, the developer only need to specify which is the web page to render. Therefore contents has to be enclosed into web pages to be displayed to users. At this point, our framework allows the visualization of contents into a window on the tabletop.

Contents can be arranged (and personalized, e. g. using a particular layout) using the CSS standard language like what happens for web sites. But the provided interaction is very poor, since the user can touch the interface, but the touch is interpreted like a movement of a mouse pointer. No gestures like pinch, rotation or drag are supported, but only tap and double tap.

Since people do not use their hand and fingers like a mouse pointer, we need the *Touch Manager* component to manages concurrent touches and gestures of many users. This software component manages portability of touches and gestures recognition and implements the TUIO protocol (Kaltenbrunner et al., 2013) which allows the transmission of an abstract description of interactive surfaces, including touch events and tangible object states. This protocol encodes control data from a tracker application (e.g. based on computer vision) and sends it to any client application that is capable to decode the protocol. Technically TUIO is based on Open Sound Control (OSC) - an emerging standard for interactive environments not only limited to musical instrument control - and can be therefore easily implemented on any platform that supports OSC.

The recognition of the gestures is managed extending *qTUIO* (Belleh and Blankenburgs, 2013), a library which implements a TUIO listener on a local UDP socket and forwards the events into the internal event system of Qt. *qTUIO* is able to recognize gestures, e. g., dragging of an object, made with one finger, two fingers are allowed only for the zoom in and out management. Since the user usually move windows and objects with the whole hand, *qTUIO* is only a first step through the realization of a portable software for the complete management of multi-touch interaction. For this reason, the *Touch Manager* extends this library to recognize and manage also gestures which involves more than one finger, e. g., multi-touch pan and pinch, scroll, drag and rotation using up to five fingers.

Since *Tactive* allows to launch more than one applications at the same time, another problem arise,

i. e., the management of application audio. In fact, if many applications use contemporary the audio interface, the result can be a big uproar, and it could be very difficult for the users to understand the audio messages. Consider, as an example, the case in which two users play contemporary two demonstrative videos, what happens is that the audio messages are overlapped and none of the users is able to easily follow the video. The situation is even worse when dealing with more users.

For this reason, *Tactive* implements the component called *Audio Manager*, which is able to manage contemporary audio. Audio messages are classified by the content editor according to their nature, i. e. soundtrack or spoken comment. More soundtracks can play together, two spoken comments cannot, so one of the two audio (and video if it is the audio comment of a video) is suspended till the end of the first one. To decide which audio is paused, the *Audio Manager* allows to define priority classes, or use a first-in, first-served policy if no priority was defined by the content editor.

3.2 Communication between Different Windows

An important component of our architecture is the *Windows Manager*. Given the dimension of the tabletop, concurrent interactions by more than one user is an important issue to consider. As an example, the users can compete for space on the surface. For these reasons, when a new window is opened (even by a new user or not), this operation can require the resize of all the other windows already present on the table. Otherwise, actions from a particular user may affect the behavior of the windows of other users. To allow the easy implementation of applications with this kind of features, *Tactive* implements a windows manager and communication protocol between windows provided by the *Message Dispatcher*.

Let us consider as example, an application with a map, e. g., a map of a city with the list of its museums, or a map of an exhibition with the position of the stands. The map can be rendered with HTML5 on a *WebView* (see Figure 3). If the user touches a museum or a stand the application opens a new window, with the web site of (or a page dedicated to) the museum/stand, and the user can interact with this window, resize it, or move across the table. If the user touches the “go to the map” button on the new windows, the initial window with the map is moved over the current window of the user. Figure 3 shows a screenshot from an application developed for a local fair.

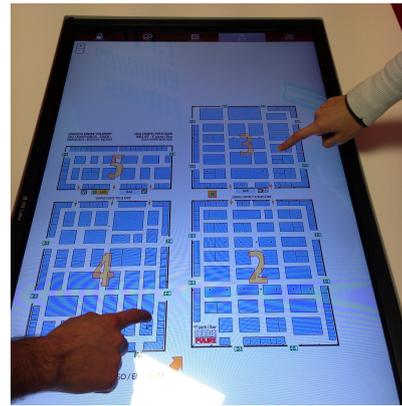


Figure 3: Screenshot from an application developed for a local exhibition.

To implement this behavior, a communication protocol between windows has been developed. The communication protocol allows the developer to change the content or the behavior of a window on the base of the behavior, or user interaction with, another windows. Each *WebView* communicates with the software layer *Tactive*, which acts as a windows manager. We need a windows manager instead of a simple communication protocol between windows, widgets or *WebViews* because only the windows manager knows how many windows are currently open in the surface, where they are, and how they are interacting with the user, each window knows only the information about itself, and nothing about the other. Moreover, the use of a windows manager allows an easy recover from the failure of a single window, since the manager records a set of information for each window and is able to stop, suspend or restart it.

Tactive implements the windows management using the C++ language to address performance issues. Moreover, it offers to developers of multi-touch application a Javascript API to manage events triggered by *Tactive* inside their web applications which use our framework to work on multi-touch interactive surfaces. The Javascript API allows to enlarge, resize, minimize, close or move a window, in response to a user interaction, also on other windows.

Moreover, using this API, it is possible to send a message to a widget active on another window through the *Message Dispatcher*. Consider as example an *advergame*: the user gains coins to play with a slot machine, answering to a questionnaire. When he/she completes the questionnaire, the window with the questions sends a message to the slot machine, enabling the user to play. This communication between windows is enabled by the Javascript API, which is used to compose the message and trigger the event through the *Message Dispatcher* to the



Figure 4: Screenshot from an application developed for doctors training.

Windows Manager, which is in charge of triggering the right response to the right window.

4 CASE STUDIES AND DISCUSSION

The framework *Tactive* has been used to develop six applications in completely different contexts, ranging from fair exposition to the launch of a new product. In this section we describe two success stories and we report some data about how the use of this framework deeply impacts the development of a multi-touch application.

The first success story is an application to improve learning activities developed for a local company. The context of use was the training of physicians. The application puts around a table four physicians, two per side. Each physicians has different materials and documents, i. e., medical records, laboratory diagnosis, x-rays, etc, about a single patient with a particular disease. No physicians has enough material to understand which is the disease which affects the patient without the help of data held by other doctors.

Figure 4 shows a screenshot of the interface. Different content is delivered to each workspace. The goal is to improve communication strategies and the ability to work together of the physicians. The doctors can create new windows to share the content, can drag the window around the table surface, rotate, zoom in and out to better understand a picture, e.g., an x-ray, or a video, e. g., an ultrasound scan. When a doctor puts in common his own material dragging it on the center of the table, the other windows are minimized, to better focus the other doctors' attention on that particular medical data.

The application was created using *Tactive*, therefore the developer only needs to assemble the content into web pages. The Javascript API was used to implement the communication between windows, i. e.,



Figure 5: Screenshot from an application developed for car market.

to minimize all the windows when a physician puts some data on the center of the table.

Thanks to our software layer, the development process is reduced to content creation which requires 45 man-days of a developer for its realization. The development of the same application using the C++ language on a TouchWindow Slice Table Multi-Touch (Touchwindow S.r.l., 2013) required one man-year, therefore our framework allows to save about 86% of time² as reported in Table 1.

The same application was used during 15 different one-day courses for physicians, using the same structure, and changing only the content, i. e., the text in the web pages, but not the structure of the pages. This adaptation process required only one day of work of a web content editor. Moreover, the application can run on any tabletop, independently from the operating system³ or the size of the surface.

The second case study is an application developed for the launch of a new product of a leading company in the car market. In this case, the application was used by a single speaker who, during his presentation, switched between an interactive slideshow, several videos and some online demos on a web site. Figure 5 shows the menu which allows to choose a video for the presentation.

The main issue for this application was to mix both off-line and online content: the “traditional” software building blocks used for tabletop UI would have required to develop the application from scratch, loading it with the off-line content (videos and slideshows) and linking online content into a webview or a browser. Such application would have required four weeks of a FTE (Full Time Equivalent) software developer, and an implementation using Flash would have required ten man-days, as reported in Table 1.

²This information has been extracted from a previous realization of the same application, which was not independent from the chosen hardware.

³Microsoft Windows 7 or superior, Linux and Apple iOS Lion are supported.

Table 1: Impact of the use of *Tactive* in the developing time for the 6 application developed using this framework. The developing time is expressed in man-days. We suppose that a man-week is equal to 5 man-days, a man-month correspond, on average, to 20 man-days, and finally, a man-year corresponds, on average, to 220 man-days.

| App | Tactive | Flash | Saving | C++ | Saving |
|---------------------|---------|-------|--------|-----|--------|
| Success Story 1 | 30 | – | – | 220 | 86% |
| Success Story 2 | 3 | 10 | 70% | 20 | 85% |
| Sculptor Exhibition | 3 | 10 | 70% | 15 | 80% |
| Innovation Festival | 5 | 20 | 75% | 60 | 91% |
| Job Event | 2 | 5 | 60% | 20 | – |
| Learning App | 2 | 5 | 60% | 20 | – |

Using the *Anytable* framework, any piece of content was linked into a different web page and published online, included the main menu page: the overall activity required 3 days of a FTE web developer, therefore it saves 85% of time with respect to an implementation using a native SDK, and 70% of time with respect to Flash implementation.

The framework has been used to implement other four applications, for a sculpture exhibition, two fairs and another type of application for learning with a different interaction with the users. Table 1 reports the required time to implement these applications using our framework. These results are compared with the estimated time required to develop the same applications using Flash and C++ language with a native SDK solution. This information has been collected from quotations that have been made during the sale phases of the final product to the customer.

We can see that our framework allows to save between 60% and 75% of developing time respect to Flash implementation. This important range of percentage rises to 80% and 91% for applications developed using C++ language and a native SDK. It is easy to note that this saving is higher for complex applications.

Although this important result in terms of time saving, our framework introduces also some drawbacks. In particular, to allow independence from the underlining hardware, we abstract from its characteristic and we implement a software layer which is able to operate with any tabletop. This means that *Tactive* defines a set of functions common to all tabletop solutions, and does not consider features which are available only on a particular hardware configuration: this choice limits the expressiveness of *Tactive*, which does not allow to use manufacturer-specific features in applications development. However, further development of HTML5 API will be considered in the future release of our software in order to lower this limitation.

5 CONCLUSION

In this paper we present *Tactive*, a software layer for fast development of *portable* applications for multi-touch interactive tabletops. The framework is based on modern web technologies and its core unit is developed using the C++ language.

The novelty of our approach consists in three points:

- the development of a framework for the creation of application for multi-touch surfaces which are independent from the hardware and software equipment;
- the possibility to use (and possible re-use) web pages decreases the time spent to develop the multi-touch applications and does not require to learn any new technology. Our experiments shows that *Tactive* allows an important reduction in time needed for development, between 60% and 91%;
- finally, no other software framework provides an easy communication between different windows of the same applications.

Moreover, our framework extends the qTUIO library to manage the recognition of gestures made with up to five fingers.

Future works will be dedicated to the implementation of an API to manage Near Field Communication (NFC). NFC is a technology that provides short-range (up to a maximum of 10 cm) and bi-directional wireless connectivity. The idea is to save the state of the user, in term of opened documents and windows, and which is the window currently active, and to re-create the entire workspace at the correct state, every time that user approaches the system.

REFERENCES

- Anthony, L., Brown, Q., Nias, J., Tate, B., and Mohan, S. (2012). Interaction and recognition challenges in interpreting children’s touch and gesture input on mo-

- bile devices. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces*, ITS '12, pages 225–234.
- Apache Software Foundation (2013). Phonegap, <http://phonegap.com/>.
- Belleh, W. and Blankenburgs, M. (2013). qTUIO Library. <http://qtuiio.sirbabyface.net/>.
- Correia, N., Mota, T., Nóbrega, R., Silva, L., and Almeida, A. (2010). A multi-touch tabletop for robust multimedia interaction in museums. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, pages 117–120.
- Forlines, C., Wigdor, D., Shen, C., and Balakrishnan, R. (2007). Direct-touch vs. mouse input for tabletop displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 647–656.
- Gaggi, O. and Regazzo, M. (2013). An environment for fast development of tabletop applications. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '13, pages 413–416.
- Geller, T. (2006). Interactive tabletop exhibits in museums and galleries. *IEEE Comput. Graph. Appl.*, 26(5):6–11.
- Hesselmann, T., Boll, S., and Heuten, W. (2011). Sciva: designing applications for surface computers. In *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering interactive computing systems*, EICS '11, pages 191–196.
- Ideum (2013). Gestureworks Core. <http://gestureworks.com/pages/core-home>.
- Kaltenbrunner, M., Bovermann, T., Bencina, R., and Costanza, E. (2013). TUIO Framework. <http://www.tuio.org/>.
- Luyten, K., Vanacken, D., Weiss, M., Borchers, J., Izadi, S., and Wigdor, D. (2010). Engineering patterns for multi-touch interfaces. In *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems*, EICS '10, pages 365–366.
- Microsoft (2013a). Surface 2.0 SDK. <http://msdn.microsoft.com/en-us/library/ff727815.aspx>.
- Microsoft (2013b). Walkthrough: Creating Your First Touch Application. <http://msdn.microsoft.com/en-us/library/ee649090.aspx>.
- Nielsen, M., String, M., Moeslund, T., and Granum, E. (2004). A procedure for developing intuitive and ergonomic gesture interfaces for hci. In Camurri, A. and Volpe, G., editors, *Gesture-Based Communication in Human-Computer Interaction*, volume 2915 of *Lecture Notes in Computer Science*, pages 409–420. Springer Berlin Heidelberg.
- Pedrosa, D., Guimarães, R. L., da Graça Pimentel, M., Bulterman, D. C. A., and Cesar, P. (2013). Interactive coffee table for exploration of personal photos and videos. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, SAC '13, pages 967–974.
- Qin, Y., Liu, J., Wu, C., and Shi, Y. (2012). uEmergency: a collaborative system for emergency management on very large tabletop. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces*, ITS '12, pages 399–402.
- Rick, J., Marshall, P., and Yuill, N. (2011). Beyond one-size-fits-all: how interactive tabletops support collaborative learning. In *Proceedings of the 10th International Conference on Interaction Design and Children*, IDC '11, pages 109–117.
- Seto, M., Scott, S., and Hancock, M. (2012). Investigating menu discoverability on a digital tabletop in a public setting. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces*, ITS '12, pages 71–80.
- SMART Technologies (2013). SMART Table SDK. <http://downloads01.smarttech.com/media/products/sdk/smart-table-sdk-summary.pdf>.
- Toffanin, P. (2013). Glassomium Project. <http://www.glassomium.org/>.
- Touchwindow S.r.l. (2013). Slice Table Multi-Touch <http://www.touchwindow.it/en/slice-multi-touch-table.php/>.
- Unedged (2013). Arena Multitouch Platform. <http://arena.unedged.com/>.
- Urakami, J. (2012). Developing and testing a human-based gesture vocabulary for tabletop systems. *Human Factors*, 54(4):636–653.
- Wigdor, D., Fletcher, J., and Morrison, G. (2009). Designing user interfaces for multi-touch and gesture devices. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 2755–2758.

On Metrics for Measuring Fragmentation of Federation over SPARQL Endpoints

Nur Aini Rakhmawati, Marcel Karnstedt, Michael Hausenblas and Stefan Decker

INSIGHT Centre, National University of Ireland, Galway, Ireland
{f_author, s_author}@deri.org

Keywords: Linked Data, Data Distribution, Federated SPARQL Query, SPARQL Endpoint.

Abstract: Processing a federated query in Linked Data is challenging because it needs to consider the number of sources, the source locations as well as heterogeneous system such as hardware, software and data structure and distribution. In this work, we investigate the relationship between the data distribution and the communication cost in a federated SPARQL query framework. We introduce the spreading factor as a dataset metric for computing the distribution of classes and properties throughout a set of data sources. To observe the relationship between the spreading factor and the communication cost, we generate 9 datasets by using several data fragmentation and allocation strategies. Our experimental results showed that the spreading factor is correlated with the communication cost between a federated engine and the SPARQL endpoints. In terms of partitioning strategies, partitioning triples based on the properties and classes can minimize the communication cost. However, such partitioning can also reduce the performance of SPARQL endpoint within the federation framework.

1 INTRODUCTION

Processing a federated query in the Linked Data is challenging because it needs to consider the number of the sources, the source locations and heterogeneous system such as the hardware, the software and the data structure and the distribution. A federated SPARQL query can be easily formulated by using the `SERVICE` keyword. Nevertheless, determining the datasource address that follows `SERVICE` keywords can be an obstacle in writing a query because prior knowledge data is required. To address this issue, several approaches (Rakhmawati et al., 2013) have been developed with the objective of hiding `SERVICE` keyword and data sources location from the user. In these approaches, the federated engines receive a query from the user, parse the query into sub queries, decide the location of each sub query and distribute the sub queries to the relevant sources. A sub query can be delivered to more than one data source if the desired answer occurs in the multiple sources. Thus, the distribution of the data can affect the federation performance (Rakhmawati and Hausenblas, 2012). As an example, consider two datasets shown in Figure 1. Each dataset contains a list of personal information using the FOAF(<http://xmlns.com/foaf/spec/>) vocabulary. If the user asks for the list of all person names, the federated engine must send a query to all data-

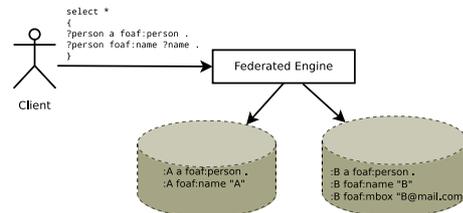


Figure 1: Example of Federated SPARQL Query Involving Many Datasets.

sources. Consequently, the communication cost between the federated engine and data sources would be expensive.

In this study, we investigate the effect of data distribution on the federated engine performance. We propose two composite metrics to calculate the presence of classes and properties across datasets. These metrics can provide insight into the data distribution in the dataset which ultimately, it can determine the communication cost between the federated engine and SPARQL Endpoints. In order to evaluate our metrics, we use several fragmentation and allocation strategies to generate different shapes of data distribution. After that, we run a static query set over those data distributions. Our data distribution strategies could be useful for benchmarking and controlled systems such as organization system, but they can not address the problem in the federated Linked

Open Data environment because the Linked Data publisher has the power to control the dataset generation. The existing evaluations for assessing the federation over SPARQL endpoints (Montoya et al., 2012; Schwarte et al., 2012) usually run their experiment over different datasets and different query sets. In fact, the performance of the federated engine is influenced by both dataset and query set. As a result, the performance results may vary. For benchmarking, a better comparison of federated engines performance can be made with either static query sets over different datasets or static dataset with various query sets.

We only perform our observation on federation over SPARQL endpoints. Query with a SERVICE keyword is also out of the scope of our study because the query only goes to the specified source. In other words, the data distribution does not influence the performance of the federation engine in that query. Our contributions can be stated as follows: 1) We investigate the effects of data fragmentation and allocation on the communication cost of the Federated SPARQL query. 2) We introduce the spreading factor as a metric for calculating the distribution of data across a dataset. In addition, we present the relationship between the spreading factor and the communication cost of federated SPARQL queries. 3) Lastly, we create datasets for evaluating the spreading factor metric drawing from the real datasets. In particular, we provide datasets and a dataset generator that can be useful for benchmarking purpose.

2 RELATED WORKS

Primitive data metrics such as the number of triples, the number of literals are not sufficiently representative to reveal the essential characteristics of the datasets. Thus, Duan (Duan et al., 2011) introduced a structuredness notion. Since this notion is applied to a single RDF repository, it is not suitable for federated SPARQL queries which should consider the data allocation in each repository as well as the number of data sources involved in the dataset.

There are several data partitioning approaches for RDF data clustering repository such as vertical partitioning (Abadi et al., 2007) and Property Table partitioning (Huang et al., 2011). However, the communication in the RDF data clustering is totally different than the communication in the federated SPARQL query. In data clustering, several machines need to communicate with each other in order to execute a query, whereas in the federated SPARQL query, there is no interaction amongst SPARQL endpoints. The mediator has a role to communicate to each

SPARQL endpoint during query execution in the federated SPARQL query. Nevertheless, we apply RDF data clustering strategies to generate the datasets for evaluation.

The existing evaluations of the federation frameworks used data partitioning in their experiment by adopting data clustering strategies. Prasser (Prasser et al., 2012) implemented three partitions: naturally-partitioned, horizontally-partitioned and randomly-partitioned. Fedbench (Schmidt et al., 2011) divided the SP2B (Schmidt et al., 2009) dataset into several partitions to run one of their evaluations. Our prior work (Rakhmawati and Hausenblas, 2012) observed the impact of data distribution on federated query execution which particularly focus on the number of sources involved, the number of links and the populated entities in several sources. In this work, we extend our previous evaluation by implementing more data partitioning schemes and we investigate the effect of the distribution of classes and properties throughout the dataset partitions on the performance of federated SPARQL query.

3 SPREADING FACTOR OF DATASET

Federated engines generally use a data catalogue to predict the most relevant sources for a sub query. The data catalogue mostly consists of a list of predicates and classes. Apart from deciding the destination of the sub queries, a data catalogue can help federated engine generate set of query execution plans. Hence, we consider computing the Spreading factor of dataset to analyse the distribution of classes and properties throughout the dataset. We initially define the dataset used in this paper as follows:

Definition 1. *Dataset D is a finite set of data sources d . In the context of federation over SPARQL endpoints, d denotes a set of triple statements t that can be accessed by a SPARQL endpoint. For each SPARQL endpoint, there exists multiple RDF graphs.*

In our work, we ignore the existence of graphs, because we are only interested in the occurrences of properties and classes in the SPARQL endpoint.

Definition 2. *Let U be the set of all URIs, B be the set of all BlankNodes, L be the set of all Literals, then a triple $t = (s, p, o) \in (U \cup B) \times U \times (U \cup L \cup B)$ where s is the subject, p is the predicate and o is the object of triple t .*

Later on, we determine the property and the class in the dataset as follows:

Definition 3. Suppose d is a datasource in the dataset D , then the set $P_d(d, D)$ of properties p in the source d is defined as $P_d(d, D) = \{p | \exists (s, p, o) \in d \wedge d \in D\}$ and the set $P(D)$ of properties p in the dataset D is defined as $P(D) = \{p | p \in P_d(d, D) \wedge d \in D\}$

Definition 4. Suppose d is a datasource in the dataset D , then the set $C_d(d, D)$ of classes c in the source d is defined as $C_d(d, D) = \{c | \exists (s, r, t, c) \in d \wedge d \in D\}$ and the set of classes c in the dataset D is defined as $C(D) = \{c | c \in C_d(d, D) \wedge d \in D\}$

Given two datasets $D = \{d_1, d_2\}$ as shown in Figure 1. Then $P_d(d_1, D) = \{\text{rdf:type, foaf:name}\}$, $P_d(d_2, D) = P(D) = \{\text{rdf:type, foaf:name, foaf:mbox}\}$ and $C_d(d_1, D) = C_d(d_2, D) = C(D) = \{\text{foaf:person}\}$.

3.1 Spreading Factor of Dataset

With the above definitions of class, property and dataset, now we can describe how we calculate the spreading factor. The spreading factor of the dataset is based on whether or not classes and properties occur. Note that, we do not count the number of times a class and property that are found in the source d because the federated engine usually relies on the presence of property in order to predict the data location of a sub query. Given dataset D that contains a set of datasets d , the normalizing number of occurrences of properties in the Dataset D ($OCP(D)$) is calculated as follows: $OCP(D) = \frac{\sum_{d \in D} |P_d(d, D)|}{|P(D)| \times |D|}$ And the normalizing number of occurrences of classes in Dataset D ($OCC(D)$) is computed as $OCC(D) = \frac{\sum_{d \in D} |C_d(d, D)|}{|C(D)| \times |D|}$

$OCP(D)$ and $OCC(D)$ have a range value from zero to one. Inspired by the *F-Measure* function, we combine $OCP(D)$ and $OCC(D)$ into a single metric which is called the Spreading Factor $\Gamma(D)$ of the dataset D . $\Gamma(D) = \frac{(1+\beta^2)OCP(D) \times OCC(D)}{\beta^2 \times OCP(D) + OCC(D)}$ where $\beta = 0.5$

We assign $\beta = 0.5$ in order to put more stress on properties than classes. The intuition is that the highest number of the query pattern delivered to SPARQL endpoint mostly contains constant predicates (Arias et al., 2011). Moreover, the number of distinct properties in the dataset is usually higher than the number of distinct classes in the dataset. The high Γ value indicates that the class and properties are spread out over the dataset.

Look back at our previous example in which we define $P_d(d_1, D)$, $P_d(d_2, D)$, $P(D)$, $C(D)$, $C_d(d_1, D)$, $C_d(d_2, D)$, then we can calculate $OCP(D) = \frac{2+3}{3 \times 2} = 0.833$ and $OCC(D) = \frac{1+1}{1 \times 2} = 1$. Finally, we obtain $\Gamma(D) = 1.172$

3.2 Spreading Factor of Dataset associated with the Queryset

The spreading factor of a dataset reveals how the whole of classes and properties are distributed over the dataset. However, a query only consists of partial properties and classes in the dataset. Thus, it is necessary to quantify the spreading factor of the dataset with respect to the queryset.

Definition 5. A query consists of set of triple patterns τ which is formally defined as $\tau(s, p, o) \in (U \cup V) \times (U \cup V) \times (U \cup L \cup V)$ where V is a set of all variables.

Given a queryset $Q = \{q_1, q_2, \dots, q_n\}$, the Q-spreading factor γ of dataset D associated with queryset Q is computed as $\gamma(Q, D) = \sum_{q \in Q} \frac{\sum_{\tau \in q} OC(\tau, D)}{|Q|}$ where the occurrences of class and property for τ is specified as

$$OC(\tau, D) = \begin{cases} \frac{ofd(o, D)}{|D|} & \text{if } p_\tau \text{ is rdf:type} \\ & \wedge o_\tau \notin V \\ \frac{pfd(p, D)}{|D|} & \text{if } p_\tau \text{ is not rdf:type} \\ & \wedge p_\tau \notin V \\ \frac{\sum_{d \in D} |P_d(d, D)|}{|D|} & \text{otherwise} \end{cases}$$

$ofd(o, D)$ denotes the occurrences of object o in the dataset D and $pfd(p, D)$ denotes the occurrences of predicate p in the dataset D which can be calculated as follows: $ofd(o, D) = \sum_{d \in D} ofd(o, d, D)$ The occurrences of object o in the source d can be explained as follows:

$$ofd(o, d, D) = \begin{cases} 1 & \text{if } o \in C_d(d, D) \\ 0 & \text{otherwise} \end{cases}$$

$pfd(p, D) = \sum_{d \in D} pfd(p, d, D)$ The occurrence of predicate p in the source d can be obtained from the following formula:

$$pfd(p, d, D) = \begin{cases} 1 & \text{if } p \in P_d(d, D) \\ 0 & \text{otherwise} \end{cases}$$

Consider an example, given a query and a dataset as shown in Figure 1, then $OC(?person \text{ a foaf:person}, D) = 1$ and $OC(?person \text{ foaf:name } ?name, D) = 1$ because foaf:person and foaf:name are located in two data sources. As a result, the q-Spreading factor $\gamma(Q, D)$ is $\frac{1+1}{1} = 2$

4 EVALUATION

We ran our evaluation on an Intel Xeon CPU X5650, 2.67GHz server with Ubuntu Linux 64-bit installed as

Listing 1: Dailymed Sample Triples.

```

dailymeddrug:82 a dailymed:drug
dailymeddrug:82 dailymed:activeingredient dailymeding:
  Phenytoin
dailymeddrug:82 rdfs:label "Dilantin-125_(Suspension)"

dailymeddrug:201 a dailymed:drug
dailymeddrug:201 dailymed:activeingredient dailymeding:
  Ethosuximide
dailymeddrug:201 rdfs:label "Zarontin_(Capsule)"

dailymedorg:Parke-Davis a dailymed:organization
dailymedorg:Parke-Davis rdfs:label "Parke-Davis"
dailymedorg:Parke-Davis dailymed:producesDrug
  dailymeddrug:82
dailymedorg:Parke-Davis dailymed:producesDrug
  dailymeddrug:201

dailymeding:Phenytoin a dailymed:ingredients
dailymeding:Phenytoin rdfs:label "Phenytoin"

dailymeding:Ethosuximide a dailymed:ingredients
dailymeding:Ethosuximide rdfs:label "Ethosuximide"

```

the Operating System and Fuseki 1.0 as the SPARQL Endpoint server. For each dataset, we set up Fuseki on different ports. We re-used the query set from our previous work (Rakhmawati and Hausenblas, 2012). We limited the query processing duration to one hour. Each query was executed three times on two federation engines, namely SPLENDID (Görlitz and Staab, 2011) and DARQ (Quilitz and Leser, 2008). These engines were chosen because SPLENDID employs VoID (<http://www.w3.org/TR/void/>) as data catalogue that contains a list of predicates and entities, while DARQ has a list of predicates which is stored in the Service Description (<http://www.w3.org/TR/sparql11-service-description/>). Apart from using VoID, SPLENDID also sends a SPARQL ASK query to determine whether or not the source can potentially return the answer. We explain the details of our dataset generation and metrics as follows:

4.1 Data Distribution

To determine the correlation between the communication cost of the federated SPARQL query and the data distribution, we generate 9 datasets by dividing the Dailymed (<http://wifo5-03.informatik.uni-mannheim.de/dailymed/>) into three partitions based on following strategies:

4.1.1 Graph Partition

Inspired by data clustering for a single RDF storage (Huang et al., 2011), we performed graph partition over our dataset by using METIS (Karypis and Kumar, 1998). The aim of this partition scheme is to reduce the communication needed between machines during the query execution process by storing the connected components of the graph in the same machine. We initially identify the connections of subject and

object in different triples. We only consider the URI object which is also a subject in other triples. Intuitively, the reason is that the object which appears as the subject in other triples can create a connection if the triples are located in different dataset partitions. $V(D)$ denotes the set of pairs of subject and object that are connected in the dataset D which can be formally specified as $V(D) = \{(s, o) | \exists s, o, p, p' \in U : (s, p, o) \in D \wedge (o, p', o') \in D'\}$. We assign a numeric identifier for each $s, o \in V(D)$. After that, we create a list of sequential adjacent vertexes for each vertex then uses it as input of METIS API. Run METIS to divide the vertexes and get a list of the partition number of vertexes as output. Finally, we distribute each triple based on the partition number of its subject and object. Consider an example, given Listing 1 as a dataset sample, then

$$V(D) = \{(dailymeddrug:82, dailymeding:Phenytoin), (dailymeddrug:201, dailymeding:Ethosuximide), (dailymedorg:Parke-Davis, dailymeddrug:82), (dailymedorg:Parke-Davis, dailymeddrug:201)\}$$

Starting an identifier value from one and increment the identifier later, we set the identifier for `dailymeddrug:82` = 1, `dailymeding:Phenytoin` = 2, `dailymeddrug:201` = 3, `dailymeding:Ethosuximide` = 4 and `dailymedorg:Parke-Davis` = 5. After that, we can create list of sequential adjacent vertexes $V(D)$ is $\{(2, 5), 1, (4, 5), 3, (1, 3)\}$. Suppose that we divide the sample of dataset into 2 partitions, then the output of METIS partition is $\{1, 1, 2, 2, 1\}$ where each value is the partition number for each vertex. According to the METIS output, we can say that `dailymeddrug:82` belongs to partition 1, `dailymeding:Phenytoin` belongs to partition 1, `dailymeddrug:201` belongs to partition 2 and so on. In the end, we have two following partitions:

Partition 1: all triples that contain `dailymeddrug:82`, `dailymeding:Phenytoin` and `dailymedorg:Parke-Davis`
 Partition 2: all triples that contain `dailymeddrug:201` and `dailymeding:Ethosuximide`

4.1.2 Entity Partition

The goal of this partition is to distribute the number of entities evenly in each partition. Different classes can be located in a single partition. However, the entities of the same class should be grouped in the same partition until the number of entities reaches the maximum number of entities for each source. We initially create a list of the subjects along with its class ($E(D)$). The set $E(D)$ of pairs of subject and its class in the dataset D is defined as $E(D) = \{(s, o) | \exists (s, rdftype, o) \in D\}$

Then, we sort $E(D)$ by its class o and store each pair of the subject and object in a partition until the number of pairs of subject and object equals to the total pairs of subject and object divided by the number of partitions. After that, we distribute the remainders of triples in the dataset based on the subject location. Given Listing 1 as a dataset sample, then

```
 $E(D)=\{(\text{dailyeddrug:82,dailyed:drug}),(\text{dailyeddrug:201},\text{dailyed:drug}),(\text{dailyedorg:Parke-Davis,dailyed:organization}),(\text{dailyeming:Phenytoin,dailyed:ingredients}),(\text{dailyeming:Ethosuximide,dailyed:ingredients})\}$ 
```

Suppose that we split the dataset into two partitions, then the maximum number of entities for each partition is $\frac{|E(D)|}{\text{numberofpartitions}} = \frac{5}{2} = 3$ (ceiling 2.5). We place `dailyeddrug:82`, `dailyeddrug:201` and `dailyedorg:Parke-Davis` in the partition 1 and store the remainders of entities in the partition 2. As the final step, we distribute the related triples based on its subject partition number.

4.1.3 Class Partition

Class Partition divides the dataset based on its classes. The related triples that belong to one entity are placed in the same machine. To begin with, we also create $E(D)$ which was used in Entity partition. Later, we distribute each triple based on the subject class. Like our previous entity partition example, we do the same step to generate $E(D)$. However, in the class partition, we divide the dataset to three partitions since we have three classes (`dailyed:drug`, `dailyed:organization`, `dailyed:ingredients`).

4.1.4 Property Partition

Wilkinson (Wilkinson, 2006) introduced a method for storing RDF data in traditional databases known as Property Table (PT). There are two types of PT partitions: Clustered Property Table and Property-class Table. In our property partition, we do not have a Property class table because we treat all properties in the same manner. We place the triples that have the same property in one data source. Because the number of properties in the dataset is generally high, we allow more than one property to be stored in the same partition as long as we get a balanced number of triples among the partitions. Firstly, we group the triples based on its property. Next, we store each group in a partition until the number of partition triples is less than or equal to the number of dataset triples divided by the number of partitions. For instance, given a dataset as shown in Listing 1, then we have four properties:

`rdf:type`, `dailyed:activeingredient`, `rdf:label` and `dailyed:producesDrug`. Suppose that we want to divide the dataset into 2 partitions, then the maximum number of triples in each partition is $\frac{\text{thenumberoftriples}}{\text{thenumberofpartitions}} = \frac{14}{2} = 7$. As the following step, we store the triples based on its property as follows: *Partition 1*: five triples with `rdf:type` property, two triples with `dailyed:activeingredient` property and *Partition 2*: five triples with `rdfs:label` property, two triples with `dailyed:producesDrug`

4.1.5 Triples Partition

The federation framework performance is influenced not only by the federated engine solely, but also depends on the SPARQL Endpoints within the federation framework. In order to keep balanced workload for SPARQL Endpoints, we split up the triples of each source evenly because LUBM (Guo et al., 2005) mentioned that the number of triples can influence the performance of a RDF repository. We created three triple partition datasets (TD , $TD2$, $TD3$). TD is obtained by partitioning the native Dailymed dataset into three parts. $TD2$ and $TD3$ are generated by picking a random starting point within the Dailymed dump file (by picking a random line number).

4.1.6 Hybrid Partition

The Hybrid Partition is a partitioning method that combines two or more previous partition strategies. For instance, if the number of triples in a class is too high, we can distribute the triples to another partition to equalize the number of triples. Since the number of triples in each dataset of the Class Distribution CD are not equal, we create HD to distribute the triples evenly. However, `rdf:type` property and `rdfs:label` property are evenly through all partitions in dataset $HD2$. This distribution is intended for balancing the workload amongst SPARQL Endpoints since those properties are commonly used in our query set.

As shown in those figures, the classes and properties are distributed over most of the partitions in the GD dataset. The PD has the lowest Spreading Factor among the dataset because each property occurs in exactly one partition and only in one partition has a set of triples that contains `rdf:type`. The dataset generation code and the generation results can be found at DFedQ github (<https://github.com/nurainir/DFedQ>)

4.2 Metrics

To calculate the communication cost of the the federated SPARQL query, we compute the data transfer

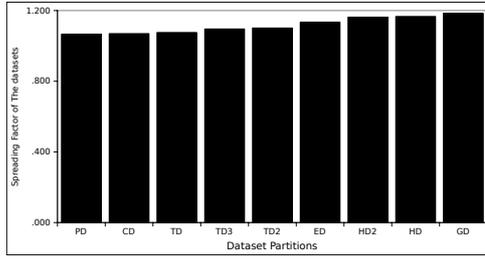


Figure 2: Spreading Factor of Dataset.

volume between the federated engine and SPARQL Endpoints. The data transfer volume includes the amount of data both sent and received by the mediator. Apart from capturing the data transmission, we also measure the requests workload (RW) during query execution. RW is calculated as $RW = \frac{RQ}{T * SS}$ where RQ refers to the number of requests sent by the federated engine to all SPARQL Endpoints, T denotes the duration between when a query is received by the federated engine and when its results starts to be dispatched to the client and SS is the number of selected sources. Furthermore, we also measure the response time that is required by a federated engine to execute a query.

For the sake of readability, we aggregate each performance metric results into a single value. In order to avoid trade-offs among queries, we assign a weight to each query using the the variable counting strategy from the ARQ Jena (Stocker and Seaborne, 2007). This weight indicate the complexity of the query based on the selectivity of the variable position and the impact of variables on the source selection process. The complexity of query can influence the federation performance. Hence, we normalize each performance metric result by dividing the metric value with the weight of the associated query. In the context of federated SPARQL queries, we set the weight of the predicate variable equals to the weight of the subject variable since most of the federated engines rely on a list of predicates to decide the data location. Note that, a triple pattern can contain more than one variable. The details of the weight of subject variable w_s , predicate variable w_p and object variable w_o for the triple pattern τ can be explained as follows:

$$w_s(\tau) = \begin{cases} 3 & \text{if the subject of triple pattern } \tau \in V \\ 0 & \text{otherwise} \end{cases}$$

$$w_p(\tau) = \begin{cases} 3 & \text{if the predicate of triple pattern } \tau \in V \\ 0 & \text{otherwise} \end{cases}$$

$$w_o(\tau) = \begin{cases} 1 & \text{if the object of triple pattern } \tau \in V \\ 0 & \text{otherwise} \end{cases}$$

Finally, we can compute the weight of query q : $weight(q) = \sum_{\forall \tau \in q} \frac{w_s(\tau) + w_p(\tau) + w_o(\tau) + 1}{MAX_COST}$ where

$MAX_COST = 8$ because if a triple pattern consists of variables that are located in all positions, the weight of the triple pattern is $8(3+3+1+1)$. By using the weight of a query, we can align the query performance results afterwards. We do not create a composite metric that combines the response time, the request workload and the data transfer, but rather we calculate each performance metric results individually. Given that Q is a set of queries q in the evaluation and that m is a set of performance metric results associated with the queryset Q , then the final metric μ for the evaluation is $\mu(Q, m) = \frac{\sum_{\forall q \in Q} \frac{mq}{weight(q)}}{|Q|}$

For instances, the query in Figure 1 has a weight $= \frac{3+1}{8} + \frac{3+1+1}{8} = 1.125$. Suppose that the volume of data transmission during this query execution is 10 Mb and we only have one query in the queryset, then $\mu(Q, m)$ can be calculated $\frac{10}{1.125} = 8.88\text{Mb}$.

5 RESULTS AND DISCUSSION

As seen in Figures 3 and 4, the data transmission between DARQ and SPARQL Endpoints is higher than the data transmission between SPLENDID and SPARQL Endpoints. However, Figures 5 and 6 show that the average requests workload in DARQ is less than the average requests workload in SPLENDID. This is because DARQ never sends SPARQL ASK queries in order to predict the most relevant source for each sub query.

Overall, data transmission increases gradually in line with the Spreading Factor of a dataset. However, the data transmission rises dramatically for GD distribution. This indicates that in the context of Federated SPARQL queries, data clustering based on its property and class is better than data clustering based on related entities such as Graph Partition. The reason behind this conclusion is that the source selection in federated query engine depends on classes and properties occurrences. Furthermore, when the federated engines generate query plans, they use optimization techniques based on the statistical predicates and classes.

Although a small Spreading Factor can minimize the communication cost, it can also reduce the SPARQL Endpoint performance. As shown in Figure 5 and 6, a small Spreading Factor can lead to the high number of requests received by SPARQL Endpoint in one second because in the property distribution, the federated engine mostly sends different query patterns to multiple datasource. Moreover, the SPARQL endpoint that stores the popular predicates such as `rdf:type` and `rdfs:label` will receive

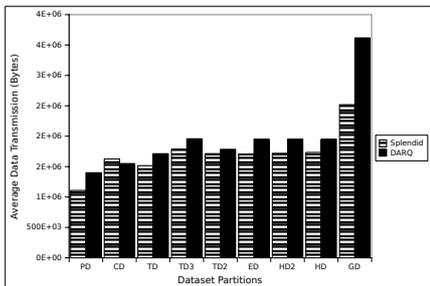


Figure 3: Average Data Transfer Volume Vs the Spreading Factor of Datasets (order by the Spreading Factor value).

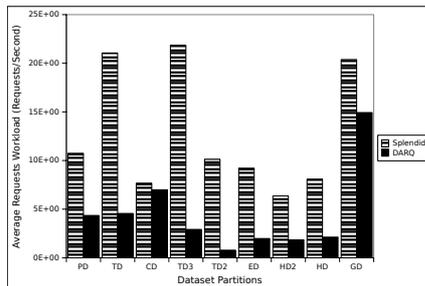


Figure 6: Average Requests Workload Vs the Q-Spreading Factor of Datasets associated with the Queryset (order by the Spreading Factor value).

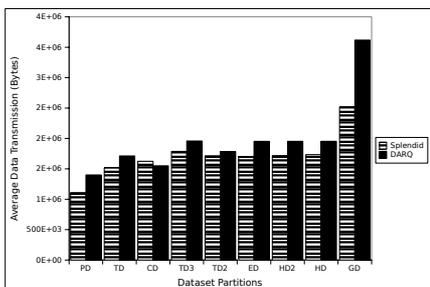


Figure 4: Average Data Transfer Volume Vs the Q-Spreading Factor of Datasets associated with the Queryset (order by the Spreading Factor value).

more requests than other SPARQL endpoints. Consequently, this such condition can lead to incomplete results because when overloaded, the SPARQL Endpoint might reject requests (e.g Sindice SPARQL endpoint(<http://sindice.com/>) only allows one client sending one query per second). Poor performance is also shown at the highest value of Spreading Factor of the dataset (GD) because the entities are spread over the dataset partitions. Hence, with the calculation of the spreading factor of the dataset, the federated engine can create a query optimization which attempts to adapt the dataset characteristic that is shown from the spreading factor value. For instance, if the dataset has too small Spreading Factor, the federated engine should maintain a timer to send several requests to the same SPARQL endpoint in order to keep the sustainability of the SPARQL endpoint as well as avoid the

incomplete answer.

6 CONCLUSION

We have implemented various data distribution strategies to partition classes and properties over dataset partitions. We introduced two notions of dataset metrics, namely the Spreading Factor of a dataset and the Spreading Factor of a Dataset associated with the query set. These metrics expose the distribution of classes and properties over the dataset partitions. Our experiment results revealed that the class and property distribution effects on the communication cost between the federated engine and SPARQL endpoints. However, it does not significantly influence the request workload of a SPARQL endpoint. Partitioning triples based on the properties and classes can minimize the communication cost. However, such partitioning can also reduce the performance of SPARQL endpoints within the federation infrastructure. Further, it can also influence the overall performance of federation framework.

In future work, we will apply other dataset partitioning strategies and use more federated query engines which have different characteristics from DARQ and SPLENDID.

ACKNOWLEDGEMENTS

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 and Indonesian Directorate General of Higher Education. Thanks to Soheila for a great discussion

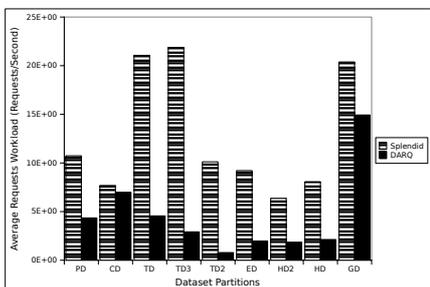


Figure 5: Average Requests Workload Vs the Spreading Factor of Datasets (order by the Spreading Factor value).

REFERENCES

- Abadi, D. J., Marcus, A., Madden, S. R., and Hollenbach, K. (2007). Scalable semantic web data management using vertical partitioning. In *Proceedings of the 33rd international conference on Very large data bases, VLDB '07*, pages 411–422.
- Arias, M., Fernández, J. D., Martínez-Prieto, M. A., and de la Fuente, P. (2011). An empirical study of real-world sparql queries. *CoRR*, abs/1103.5043.
- Duan, S., Kementsietsidis, A., Srinivas, K., and Udea, O. (2011). Apples and oranges: a comparison of rdf benchmarks and real rdf datasets. In *ACM International Conference on Management of Data (SIGMOD)*.
- Görlitz, O. and Staab, S. (2011). SPLENDID: SPARQL Endpoint Federation Exploiting VOID Descriptions. In *COLD2011*, Bonn, Germany.
- Guo, Y., Pan, Z., and Heflin, J. (2005). Lubm: A benchmark for owl knowledge base systems. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2-3):158 – 182.
- Huang, J., Abadi, D. J., and Ren, K. (2011). Scalable sparql querying of large rdf graphs. *PVLDB*, 4(11):1123–1134.
- Karypis, G. and Kumar, V. (1998). A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392.
- Montoya, G., Vidal, M.-E., Corcho, Ó., Ruckhaus, E., and Aranda, C. B. (2012). Benchmarking federated sparql query engines: Are existing testbeds enough? In *International Semantic Web Conference (2)*, pages 313–324.
- Prasser, F., Kemper, A., and Kuhn, K. A. (2012). Efficient distributed query processing for autonomous rdf databases. In *Proceedings of the 15th International Conference on Extending Database Technology, EDBT '12*, pages 372–383, New York, NY, USA. ACM.
- Quilitz, B. and Leser, U. (2008). Querying distributed rdf data sources with sparql. In *ESWC2008*, pages 524–538, Berlin, Heidelberg. Springer-Verlag.
- Rakhmawati, N. A. and Hausenblas, M. (2012). On the impact of data distribution in federated sparql queries. In *Semantic Computing (ICSC), 2012 IEEE Sixth International Conference on*, pages 255 –260.
- Rakhmawati, N. A., Umbrich, J., Karnstedt, M., Hasnain, A., and Hausenblas, M. (2013). Querying over federated sparql endpoints - a state of the art survey. *CoRR*, abs/1306.1723.
- Schmidt, M., Görlitz, O., Haase, P., Ladwig, G., Schwarte, A., and Tran, T. (2011). Fedbench: A benchmark suite for federated semantic data query processing. In *ISWC*
- Schmidt, M., Hornung, T., Lausen, G., and Pinkel, C. (2009). Sp²bench: a sparql performance benchmark. In *Data Engineering, 2009. ICDE'09. IEEE 25th International Conference on*, pages 222–233. IEEE.
- Schwarte, A., Haase, P., Schmidt, M., Hose, K., and Schenkel, R. (2012). An experience report of large scale federations. *CoRR*, abs/1210.5403.
- Stocker, M. and Seaborne, A. (2007). Arqo: The architecture for an arq static query optimizer.
- Wilkinson, K. (2006). Jena property table implementation. In *In SSWS*.

Development Process and Evaluation Methods for Adaptive Hypermedia

Martin Balík and Ivan Jelínek

*Department of Computer Science and Engineering, Faculty of Electrical Engineering, Czech Technical University
Karlovo náměstí 13, 121 35 Prague, Czech Republic
{balikm1, jelinek}@fel.cvut.cz*

Keywords: Adaptive Hypermedia, Personalization, Development Process, Software Framework, Evaluation.

Abstract: Adaptive Hypermedia address the fact that each individual user has different preferences and expectations. Hypermedia need adaptive features to provide an improved user experience. This requirement results in an increased complexity of the development process and evaluation methodology. In this article, we first discuss development methodologies used for hypermedia development in general and especially for user-adaptive hypermedia development. Second, we discuss evaluation methodologies that constitute a very important part of the development process. Finally, we propose a customized development process supported by ASF, a special framework designed to build Adaptive Hypermedia Systems.

1 INTRODUCTION

Software development is a complex process, where modeling and specification on various levels have become a necessity and a standard approach. Web-based hypermedia systems require a special attention that has led to evolution of a new line of research – Web Engineering (Deshpande et al., 2002). A number of development methodologies have been created to offer new techniques, models and notations. Additional challenges came with a new category of intelligent, user-adaptive applications.

User-adaptive systems monitor users' behavior and keep track of each individual user's characteristics, preferences, knowledge, aims, etc. Some of the systems focus on providing the user with relevant items based on the browsing history. Other systems focus mainly on improving the human-computer interactions. The collection of personal data used in the adaptation process is associated with a specific user. It is called the User Model. While modeling the adaptive system, it is necessary to separate the non-adaptive and user-specific aspects of the application.

In our work, we focus on Adaptive Hypermedia Systems (AHS). Typical adaptation techniques used in AHS are categorized as *content adaptation*, *adaptive presentation*, and *adaptive navigation* (Knutov et al., 2009). The categories overlap, as some of the techniques do not change information or the possible navigation, but only offer suggestions

to the user by changing the presentation. The design of adaptation techniques needs to be considered within the development process.

User-adaptive systems bring additional complexity into the development process and lay higher demands on system evaluation. This needs to be considered through all development phases. In order to guarantee the required behavior, we have to ensure that the system works correctly during and after adaptations (Zhang and Cheng, 2006).

Evaluation of adaptive systems is an important part of their development process and should not be underestimated. Currently, there is not much consistency in the evaluation of AHS (Mulwa et al., 2011). It is important to use an appropriate method for evaluation (Gena and Weibelzahl, 2007). Evaluation should ensure savings in terms of time and cost, completeness of system functionality, minimizing required repair efforts, and improving user satisfaction (Nielsen, 1993). AHSs are interactive, hypermedia-based systems. Usually, similar methods as in human-computer interaction (HCI) field are used. However, user-adaptive systems introduce new challenges.

The remainder of this paper is structured as follows. In Section 2, a current state of the art of development and evaluation methodologies is being reviewed. In Section 3, AHS development process is proposed and associated with the use of Adaptive System Framework. Finally, Section 4 concludes the paper by summarizing results of the research and indicates the directions of the future work.

2 RELATED WORK

In this section, we will review existing approaches used in AHS development. First, we will focus on the development methodologies mainly focused on design and system architecture. Second, we will review evaluation methodologies and problems related specifically with user-adaptive system evaluation.

2.1 AHS Development Methodologies

Similar to development of other software products, adaptive-system development needs to be based on standardized methods. For the design of hypermedia applications, several methods have been developed. In the early period of hypermedia systems, hypermedia-specific design methodologies were proposed, for example, Hypermedia Design Method (HDM) (Garzotto et al., 1993), Relationship Management Methodology (RMM) (Isakowitz et al., 1995), Enhanced Object-Relationship Model (EORM) (Lange, 1994) and Web Site Design Method (WSDM) (De Troyer and Leune, 1998). An Overview of additional and more recent development methodologies for software and Web engineering can be found in (Aragón et al., 2013; Thakare, 2012). However, the methodologies developed for hypermedia systems in general do not take into account the adaptivity and user modeling. Therefore, an extended adaptation-aware methodology is needed to improve the AHS development process.

Fig. 1 shows the typical phases of a software-development process. To abstract complex problems of the system design, models are used. The models help to create and validate the software architecture.



Figure 1: Typical phases of a software devel. process.

Model-Driven Architecture (MDA) (Miller and Mukerji, 2003) was proposed by the Object Management Group (OMG) in 2001. This architecture defines four model levels. *Computation-Independent Model (CIM)* describes behavior of the system in a language appropriate for users and business analysts. This level includes models of requirements and business models. *Platform-Independent Model (PIM)* is still independent of a specific computer technology, yet unlike the CIM it includes information essential for solving the assignment using information technologies. The PIM is usually created by computer analyst. The benefit of this level is the reusability for various implementations and platform independency.

Platform-Specific Model (PSM) combines the PIM with a particular technology-dependent solution. This model can include objects tightly related to a specific programming language environment, e.g., constructors, attribute accessors, or references to classes included in the development platform packages. The model is an abstraction of source code structure and is used as a base for implementation. *Code* is the highest level of MDA and includes the implementation of the system.

Adaptive systems usually access large information base of domain objects, and their behavior is based on information stored in the user model. Such systems are quite complex and therefore, development methodology oriented on adaptive hypermedia is needed.

Object-oriented approach in designing adaptive hypermedia systems seems to be the most appropriate. Object oriented design is best suited for systems undergoing complex transitions over time (Papaslouros and Retalis, 2002). For object-oriented software systems modeling, we have a standard, widely-adopted, formally defined language – UML (Booch et al., 1999). To be able to express a variety of system models, UML provides extension mechanisms in definition of the model elements, description of the notation and expressing semantic of models. These extensions are stereotypes, tagged values and constraints. UML stereotypes are the most important extension mechanism.

There are some projects that utilize UML modeling in the area of adaptive systems. The Munich Reference Model (Koch and Wirsing, 2001) is an extension of the Dexter model. It was proposed in the same period as the well-known Adaptive Hypermedia Application Model (AHAM) (De Bra et al., 1999) and in a similar way adds a user model and an adaptation model. The main difference between The Munich Reference Model and AHAM is that AHAM specifies an adaptation rule language, while The Munich Reference Model uses object-oriented specification. It is described with the Unified Modeling Language (UML) which provides the notation and the object-oriented modeling techniques.

Object-Oriented Hypermedia Design Method (OOHDM) (Rossi and Schwabe, 2008) is based on both HDM and the object-oriented paradigm. It allows the designer to specify a Web application by using several specialized meta-models. OOHDM proposed dividing hypermedia design into three models – a conceptual model, a navigational model and an abstract interface model. When used to design a user-adaptive application, most of the personalization aspects are captured in the conceptual model.

As an example, we can mention a class model of the user and user group models (Barna et al., 2004).

Another method to specify design of complex Web sites is WebML (Ceri et al., 2000). For the phase of conceptual modeling, WebML does not define its own notation and proposes the use of standard modeling techniques based on UML. In the next phase, the hypertext model is defined. This model defines the Web site by means of two sub-models – composition model and navigation model. Development of the presentation model defining the appearance of the Web site is the next step. Part of the data model is the personalization sub-schema. The content management model specifies how is the information updated dynamically based on user's actions. Finally, the presentation model specifies how the system has to be adapted to each user's role (Aragón et al., 2013).

For the purpose of interoperability, storage models can be represented by a domain ontology. Therefore, there is a need to represent ontology-based models in a standardized way. Researchers already identified this issue and proposed UML profile for OWL and feasible mappings, which support the transformation between OWL ontologies and UML models and vice versa (Brockmans et al., 2006). This is achieved by the UML stereotypes. Table 1 provides the mappings for the most important constructs.

Table 1: UML and OWL mappings (Brockmans, 2006).

| UML Feature | OWL Feature | Comment |
|--------------------------------------|---|--|
| class, type | class | |
| instance | individual | |
| ownedAttribute, binary association | property, inverseOf | |
| subclass, generalization, | subclass, subproperty, | |
| N-ary association, association class | class, property | Requires decomposition |
| enumeration | oneOf | |
| disjoint, cover | disjointWith, unionOf | |
| multiplicity | minCardinality, maxCardinality, FunctionalProperty, InverseFunctionalProperty | OWL cardinality restrictions declared only for range |
| package | ontology | |

Special attention should be also devoted to the development of the content of the adaptive systems. As it was observed many times – authoring of adaptive systems is a difficult task (Cristea, 2003). The adaptive-system development process can be divided into four phases: Conceptual Phase, Presentation Phase, Navigation Phase and Learning Phase (Medina et al., 2003).

During the conceptual phase, the author creates basic page elements, in the presentation phase the structure of page elements is defined, in the navigation phase the navigational map is created and in the learning phase, adaptive behavior is defined.

2.2 AHS Evaluation Methodologies

Recent research has identified the importance of user-adaptive systems evaluation. Reviews on the topic have been published by several researchers (Gena, 2005; Velsen et al., 2008; Mulwa et al., 2011; Albert and Steiner, 2011). Due to the complexity of adaptive systems, the evaluation is difficult. The main challenge lies in evaluating particularly the adaptive behavior. Evaluation of adaptive systems is a very important part of the development process. Moreover, it is necessary, that correct methods and evaluation metrics are used.

Usability is evaluated by the quality of interaction between a system and a user. The unit of measurement is the user's behavior (satisfaction, comfort) in a specific context of use (Federici, 2010). Design of adaptive hypermedia systems might violate standard usability principles such as user control and consistency. Evaluation approaches in HCI assume that the interactive system's state and behavior are only affected by direct and explicit action of the user (Paramythis et al., 2010). This, however, is not true in user-adaptive systems.

Personalization and user-modeling techniques aim to improve the quality of user experience within the system. However, at the same time these techniques make the systems more complex. By comparing the adaptive and non-adaptive versions, we should determine the added benefits of the adaptive behavior.

General (non-adaptive) interactive systems acquire from user the data strictly related to the performed task. Adaptive systems, however, require much more information. This information might not be required for the current task and can be in the current context completely unrelated. This is caused by continuous observation of the user by the system. Adaptive systems can monitor visited pages, keystrokes or mouse movement. Users can be even asked superfluous information directly. Within the evaluation process, it is challenging to identify the purpose and correctness of such a meta-information.

Important difference between evaluation of adaptive and non-adaptive systems is that evaluation of adaptive systems cannot consider the system as a whole. At least two layers have to be evaluated separately (Gena, 2005).

In the next paragraphs, we will summarize the most important methods used to evaluate adaptive hypermedia systems.

Comparative Evaluation

It is possible to assess the improvements gained by adaptivity by comparing the adaptive system with

a non-adaptive variant of the system (Höök, 2000). However, it is not easy to make such comparison. It would be necessary, to decompose the adaptive application into adaptive and non-adaptive components. Usually adaptive features are an integral part of the system, and the non-adaptive version could lead to unsystematic and not optimal results. Additionally, it might not be clear why the adaptive version is better.

In case of adaptive learning, a typical application area of adaptation, it is possible to compare the system with a different learning technology or with traditional learning methods. However, the evaluation of adaptation effects can interfere with look and feel or a novelty effect (Albert and Steiner, 2011).

Empirical Evaluation

Empirical evaluation, also known as the controlled experiment, appraises theories by observations in experiments. This approach can help to discover failures in interactive systems, that would remain uncovered otherwise. For software engineering, formal verification and correctness are important methods. However, empirical evaluation is an important complement that could contribute for improvement significantly. Empirical evaluation has not been applied for the user modeling techniques very often (Weibelzahl and Weber, 2003). However, in recent studies, the importance of this approach is pointed out (Paramythis et al., 2010). This method of evaluation is derived from empirical science and cognitive and experimental psychology (Gena, 2005). In the area of adaptive systems, the method is usually used for the evaluation of interface adaptations.

Layered Evaluation

For evaluation of adaptive hypermedia systems, usually approaches considering the system “as a whole” and focusing of an “end value” are used. Examples of the focused values are user’s performance or users’s satisfaction. The problem of this approach is, that evaluating system as a whole requires building the whole system before evaluation. This way, the evaluation is not able to guide authors in the development process. Another problem is, that the reasons behind unsatisfactory adaptive behavior are not evident.

A solution to the mentioned problems was proposed by Brusilovsky in (Brusilovsky and Sampson, 2004) as a model-based evaluation approach called *layered evaluation*. In the exemplary case, two layers were defined – user modeling layer and adaptation decision making layer. User modeling (UM) is the process, where information about user is acquired by

monitoring user-computer interaction. Adaptation decision making is a phase, where specific adaptations are selected, based on the results of the UM phase. Both processes are closely interconnected. However, when evaluating the system as a whole, it is not evident, which of the phases has been unsuccessful. This is solved by decomposing evaluation into layers and evaluating both phases separately. This has also the benefit, that results of UM process evaluation can be reused for different decision making modules.

Layered evaluation has gained a high level of attention in the adaptive hypermedia research community. That reaffirms the claim that the evaluation of adaptive systems implicates some inherent difficulties (Mulwa et al., 2011). The original idea is often used by authors to justify experimental designs of their evaluation studies.

Process-oriented Evaluation

Evaluation should be considered as an inherent part of the development cycle. Continuous evaluation should range from very early phases of the project till the end. Evaluation should start with requirements analysis and continue at the prototype level. Evaluation of initial implementations is referred as *formative evaluation*. Identifying early issues can greatly reduce development costs. The quality of the overall system is evaluated in the final phase of the development cycle and is referred a *summative evaluation*. The focus of current evaluations of adaptive systems is mostly targeted on the summative evaluation. To ensure that user’s needs are sufficiently reflected, formative evaluation must be more intensively used.

User-centered Evaluation

For adaptive systems, especially user-centered evaluation approaches are recommended (Velsen et al., 2008).

Following are the typical user-centered evaluation methods:

- **Questionnaires**

Questionnaires collect data from users by answering a fixed set of questions. They can be used to collect global impressions or to identify problems. Advantage is, that large number of participants can be accommodated (compared to interviews).

- **Interviews**

In interviews, participants are asked questions by an interviewer. Interviews can identify individual and situational factors and help explain, why a system will or will not be adopted.

- **Data Log Analysis**

The log analysis can focus on user behavior or the user performance. It is strongly advised to use this method with a qualitative user-centred evaluation.

- **Focus Groups and Group Discussions**

Groups of participants discuss a fixed set of topics, and the discussion is led by a moderator. This method is suitable for gathering a large amount of qualitative data in a short time.

- **Think-aloud Protocols**

Participants are asked to say their thoughts out loud while using the system.

- **Expert Reviews**

System is reviewed by an expert, who gives his opinion.

3 AHS DEVELOPMENT PROCESS

The development methodologies mentioned in section 2.1 were developed for non-adaptive hypermedia systems and therefore, the methodologies do not provide sufficient support for the adaptation process. By adding adaptive features, the design complexity increases. Without adequate development support, the application can become unmaintainable, or the behavior of the application can become inconsistent.

As an example of deficiency, the OOHDM methodology allows user-role-based personalization as part of the conceptual model. However, there is no clear separation of the user-adaptive behavior. Although the WebML defines an explicit personalization model for users and user-groups, it is missing means for expressing and separating various adaptation methods. Other legacy development methodologies do not consider personalization at all.

In a development methodology, two important components can be identified. One of them is the language, which can be used by a designer to model the different aspects of the system. The other component is the development process, which acts as the dynamic, behavioral part. The development process determines what activities should be carried out to develop the system, in what order and how. To specify the development process for user-adaptive hypermedia systems, we follow the model-driven architecture (MDA).

Fig. 2 depicts the MDA adopted to user-adaptive hypermedia systems engineering. The principles are visualized as a stereotyped UML activity diagram based on the diagram presented in (Koch et al., 2006). The process starts with the Computation-Independent

Model (CIM) that defines requirements models and user characteristics model. Platform-Independent Model (PIM) is divided into two segments. User Independent Model (UIM) describes the system without its adaptation features and is equivalent to the standard web engineering design methodology. Three models, based on the OOHDM, are created – conceptual model, navigational model, and abstract interface model. The other segment consists of the User Specific Model (USM). USM consists of three sub-models, that are patterned on adaptation method categories (Knutov et al., 2009) – content adaptation model, adaptive navigation model, and adaptive presentation model.

The user-specific PIM sub-models are closely related with our theoretical basis of adaptive hypermedia architecture – the Generic Ontology-based Model for Adaptive Web Environments (GOMAWE). The adaptation function, defined as a transformation between default and adapted hypermedia elements, is the basis for content adaptation. Transformations are defined by Inference Rules. Adaptive navigation defines transformations within the navigational model, and results into the Link-Adaptation Algorithms in subsequent modeling phases. Adaptive presentation is modeled as transformations within the Adaptive Hypermedia Document Template. For formal definitions of GOMAWE, see (Balík and Jelínek, 2013b).

After the models for both the user-independent and user-specific segments are separately defined, they can be transformed and merged together to form the “big picture” of the system. The next step is transforming the PIM into the Platform-Specific model (PSM). As an example, we show Java and .NET model, but there are many other possible platforms. From the PSM, a program code can be possibly generated.

While the PIM depends usually in large extent on UML and UML profiles that provide a standard abstract model notation, the PSM, on the other hand, should refer to software framework packages used to simplify the development on a specific platform. In our previous work, we have proposed a software framework intended to support the development of user-adaptive hypermedia systems. The Adaptive System Framework (ASF) (Balík and Jelínek, 2013a) defines a fundamental adaptive hypermedia system architecture and implements the most common adaptive system components.

One of the important ASF components is the user-specific data storage. The centralized user model management is beneficial for the application development. Using the adaptation manager, the user pro-

file and user model properties can be accessed from any component of the application. Another part of the data-storage layer is the rule repository. A rule-repository manager provides an interface for accessing and evaluating the inference rules. This interface can be utilized in the adaptation algorithms, e.g., the content adaptation algorithm can use conditional rules to find an alternative content for a specific user.

The design of the application core based on the ASF framework consists of the following important steps:

1. Definition of the domain objects and their relations
2. Definition of the user profile and user model attributes
3. Design of the adaptive algorithms for the desired behavior
4. Configuration of data sources
5. Binding the data results either to the application logic or directly to the adaptive UI components

All the steps are supported by the ASF framework. Based on UML model, the developer implements do-

main objects by using support classes of the framework. User data storage needs only data model specification (preferably as an ontology). Adaptive algorithms can be reused or extended. And finally, user interface components can be used to support the presentation.

The implemented user-adaptive application needs to be evaluated, and evaluation should be an integral part of the development process. Various methods mentioned in Section 2.2 can be used.

Based on the evaluation methodology proposed in (Lampropoulou et al., 2010), we use a three-phase evaluation as part of the development process. The first phase is a short empirical study, in the second phase a qualitative and quantitative measurement is performed, and finally, the third phase evaluates subjective comments of test session participants. For the purpose of AHS evaluation, we extend the second and third phases by the comparison with a non-adaptive system users control group.

Typical adaptive system evaluation is based on comparison between adaptive and non-adaptive version of the application. ASF framework is well

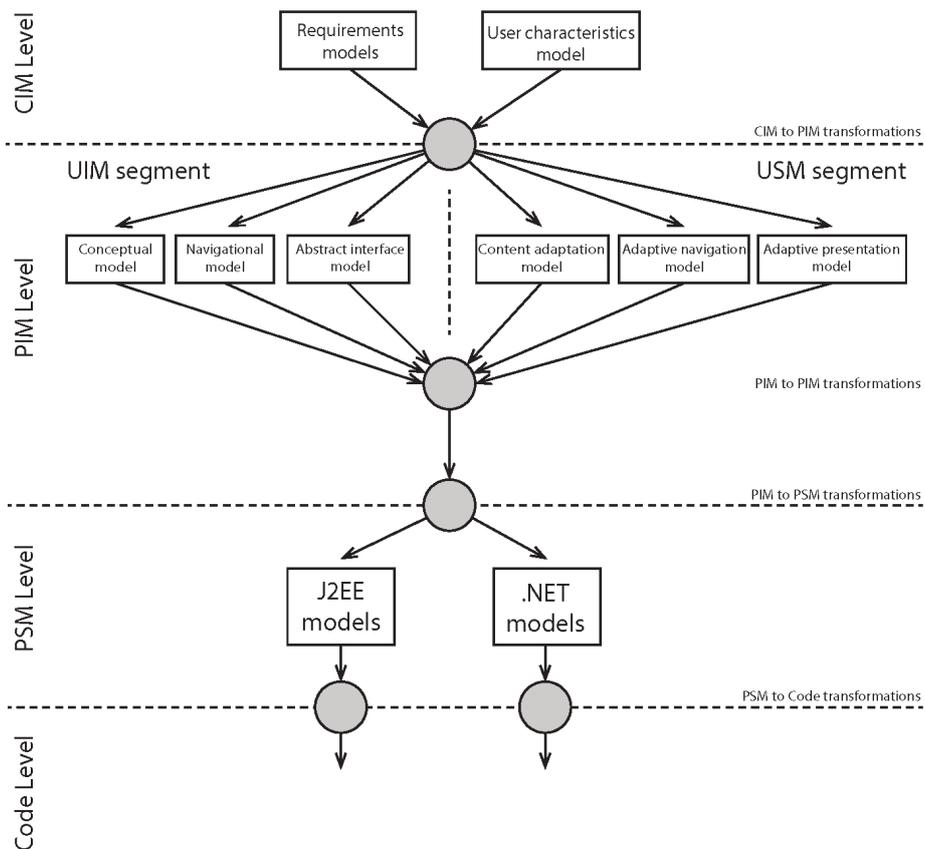


Figure 2: MDA structure for user-adaptive hypermedia systems engineering.

designed for such a comparative evaluation. Each adaptation-type algorithm strategy supports the non-adaptive algorithm version. This feature can be used as an additional user-accessible preference setting, or it can be administered for special purposes, e.g., to support the adaptation evaluation session.

To be reasonable, the evaluation needs to be performed with a representative group of users. For such purposes, adaptive educational applications, where a large amount of students can participate, is highly appropriate. Many of the typical aspects of adaptive applications can be simulated and evaluated by students. The tutorials can include theoretical tests, practical assignments, or test questions used to review the knowledge of students.

In our adaptive e-learning prototype, we focus mainly on the user-centered evaluation. In the first evaluation phase, the students were asked about their preferences regarding the online curriculum. The questions included preference of used adaptation techniques. They were also asked if their results should be available to the tutor with all details, in a form of whole class statistics, or completely hidden. In the second evaluation phase, we used a data log analysis to observe the behavior of users, progress in knowledge and selected preferences. The session with multiple students is suitable to measure the system performance, identify possible bottlenecks and compare the adaptive system with the non-adaptive alternative. The evaluation sessions are usually combined with questionnaires, where students answer questions related to the application content, and they can provide a feedback about their satisfaction or issues they encounter while using the system. This is the last of the three evaluation phases. Afterwards, all the collected data are analyzed, and the results provide a feedback for system customization and improvements.

4 CONCLUSION

Design, modeling and evaluation are fundamental steps in the development process of software products. Web technologies and requirements of personalization add more complexity into the process, and specialized methodologies are needed. In this paper, we have given an overview of existing methodologies and their use in the context of user-adaptive systems. Further, we have proposed a special methodology for adaptive hypermedia, based on MDA and OOHD. The development methodology was extended to include the aspects of user-adaptive systems. The AHS-specific methodology is important for

improving the development effectiveness and quality of the resulting product.

In our future work, we aim to use the methodology in additional prototypes' development based on ASF. We will apply the framework in different application types, and we will focus in more detail on adaptive systems' recommendation adaptation features. Further, we want to integrate the learning curriculum application with other systems and assessments used in the courses, and we want to utilize the ontology-based data maintained by the adaptive systems to exchange the user models of students.

ACKNOWLEDGEMENTS

The results of our research form a part of the scientific work of a special research group WEBING.¹

REFERENCES

- Albert, D. and Steiner, C. M. (2011). Reflections on the Evaluation of Adaptive Learning Technologies. In *Proceedings of the IEEE International Conference on Technology for Education (T4E)*, pages 295–296.
- Aragón, G., Escalona, M.-J., Lang, M., and Hilera, J. R. (2013). An Analysis of Model-Driven Web Engineering Methodologies. *Int. Journal of Innovative Computing, Information and Control*, 9(1):413–436.
- Balík, M. and Jelínek, I. (2013a). Adaptive System Framework: A Way to a Simple Development of Adaptive Hypermedia Systems. In *The Fifth International Conference on Adaptive and Self-Adaptive Systems and Applications (ADAPTIVE2013)*, pages 20–25, Valencia, Spain. IARIA.
- Balík, M. and Jelínek, I. (2013b). Generic Ontology-Based Model for Adaptive Web Environments: A Revised Formal Description Explained within the Context of its Implementation. In *Proceedings of the 13th IEEE International Conference on Computer and Information Technology (CIT2013)*, Sydney, Australia.
- Barna, P., Houben, G.-j., and Frasincar, F. (2004). Specification of Adaptive Behavior Using a General-purpose Design Methodology for Dynamic Web Applications. In *Proceedings of AdaptiveHypermedia and Adaptive Web-Based Systems (AH 2004)*, Eindhoven, The Netherlands.
- Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The Unified Modeling Language User Guide*, volume 30 of *Addison-Wesley object technology series*. Addison-Wesley.
- Brockmans, S., Colomb, R. M., Haase, P., Kendall, E. F., Wallace, E. K., Welty, C., and Xie, G. T. (2006). A model driven approach for building OWL DL and

¹Webing research group – <http://webing.felk.cvut.cz>

- OWL full ontologies. In *ISWC'06 Proceedings of the 5th international conference on The Semantic Web*, pages 187–200. Springer-Verlag.
- Brusilovsky, P. and Sampson, D. (2004). Layered evaluation of adaptive learning systems. *Int. J. Continuing Engineering Education and Life-Long Learning*, 14(4-5):402–421.
- Ceri, S., Fraternali, P., Bongio, A., and Milano, P. (2000). Web Modeling Language (WebML): a modeling language for designing Web sites. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 33(1-6):137–157.
- Cristea, A. (2003). Automatic authoring in the LAOS AHS authoring model. In *Hypertext 2003, Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems*, Nottingham, UK.
- De Bra, P., Houben, G.-J., and Wu, H. (1999). AHAM : A Dexter-based Reference Model for Adaptive Hypermedia. In *Proceedings of the ACM Conference on Hypertext and Hypermedia*, pages 147–156. ACM.
- De Troyer, O. and Leune, C. (1998). WSDM: a user centered design method for Web sites. In *Proceedings of the Seventh International WWW Conference*, volume 30, pages 85–94, Brisbane, Australia. Elsevier Science Publishers B. V.
- Deshpande, Y., Murugesan, S., Ginige, A., Hansen, S., Schwabe, D., Gaedke, M., and White, B. (2002). Web engineering. *Journal of Web Engineering*, 1(1):3–17.
- Federici, S. (2010). Usability evaluation : models, methods, and applications. In *International Encyclopedia of Rehabilitation*.
- Garzotto, F., Schwabe, D., and Paolini, P. (1993). HDM A Model-Based Approach to Hypertext Application Design. *ACM Trans. on Information Systems*, 11(1):1–26.
- Gena, C. (2005). Methods and techniques for the evaluation of user-adaptive systems. *The Knowledge Engineering Review*, 20(01):1.
- Gena, C. and Weibelzahl, S. (2007). Usability Engineering for the Adaptive Web. In Brusilovsky, P., Kobsa, A., and Nejd, W., editors, *The adaptive web*, pages 720–762. Springer-Verlag, Berlin, Heidelberg.
- Höök, K. (2000). Steps to take before intelligent user interfaces become real. *Interacting with Computers*, 12(4):409–426.
- Isakowitz, T., Stohr, E. A., and Balasubramanian, P. (1995). RMM: a methodology for structured hypermedia design. *Communications of the ACM*, 38(8):34–44.
- Knutov, E., De Bra, P., and Pechenizkiy, M. (2009). AH 12 years later: a comprehensive survey of adaptive hypermedia methods and techniques. *New Review of Hypermedia and Multimedia*, 15(1):5–38.
- Koch, N. and Wirsing, M. (2001). Software Engineering for Adaptive Hypermedia Applications? In *8th International Conference on User Modeling*, pages 1–6, Sonthofen, Germany.
- Koch, N., Zhang, G., and Escalona, M. J. (2006). Model Transformations from Requirements to Web System Design. In *Proc. Sixth Int'l Conf. Web Eng.*, pages 281–288.
- Lampropoulou, P. S., Lampropoulos, A. S., and Tsihrintzis, G. A. (2010). A Framework for Evaluation of Middleware Systems of Mobile Multimedia Services. In *The 2010 IEEE International Conference on Systems, Man, and Cybernetics (SMC2010)*, pages 1041–1045, Istanbul, Turkey.
- Lange, D. (1994). An Object-Oriented Design Method for Hypermedia Information Systems. In *Proceedings of the 27th - Hawaii International Conference on System Sciences*, volume 6, pages 336–375, Hawaii. IEEE Computer Society Press.
- Medina, N., Molina, F., and García, L. (2003). Personalized Guided Routes in an Adaptive Evolutionary Hypermedia System. In *Lecture Notes in Computer Science 2809*, pages 196–207.
- Miller, J. and Mukerji, J. (2003). MDA Guide Version 1.0. [Accessed: August 24, 2013]. [Online]. Available: http://www.omg.org/mda/mda_files/MDA_Guide_Version1-0.pdf.
- Mulwa, C., Lawless, S., Sharp, M., and Wade, V. (2011). The evaluation of adaptive and personalised information retrieval systems: a review. *International Journal of Knowledge and Web Intelligence*, 2(2-3):138–156.
- Nielsen, J. (1993). *Usability Engineering*. Academic Press, Boston, MA.
- Papasalouros, A. and Retalis, S. (2002). Ob-AHEM: A UML-enabled model for Adaptive Educational Hypermedia Applications. *Interactive educational Multimedia*, 4(4):76–88.
- Paramythis, A., Weibelzahl, S., and Masthoff, J. (2010). Layered evaluation of interactive adaptive systems: framework and formative methods. *User Modeling and User-Adapted Interaction*, 20(5):383–453.
- Rossi, G. and Schwabe, D. (2008). Modeling and implementing web applications with OOADM. In Rossi, G., Pastor, O., Schwabe, D., and Olsina, L., editors, *Web Engineering*, chapter 6, pages 109–155. Springer.
- Thakare, B. (2012). Deriving Best Practices from Development Methodology Base (Part 2). *International Journal of Engineering Research & Technology*, 1(6):1–8.
- Velsen, L. V., van der Geest, T., Klaasen, R., and Steehouder, M. (2008). User-centered evaluation of adaptive and adaptable systems: a literature review. *The Knowledge Engineering Review*, 23(3):261–281.
- Weibelzahl, S. and Weber, G. (2003). Evaluating the inference mechanism of adaptive learning systems. In *User Modeling 2003*, pages 154–162. Springer-Verlag.
- Zhang, J. and Cheng, B. H. C. (2006). Model-based development of dynamically adaptive software. In *Proceeding of the 28th international conference on Software engineering - ICSE '06*, number 1, pages 371–380, New York, New York, USA. ACM Press.

CAPTCHA and Accessibility

Is This the Best We Can Do?

Lourdes Moreno, María González and Paloma Martínez

*Computer science department, Universidad Carlos III de Madrid, Av. Universidad, 30, Leganes, Spain
{lmoreno, mgonza1, pml}@inf.uc3m.es*

Keywords: Web Accessibility, CAPTCHA.

Abstract: Web access is affected by a great amount of accessibility issues that do not allow some users to access all information presented. Therefore, Web accessibility is an important issue because everybody should access Web content independently of their access features. Among these accessibility issues, a Web content element that interferes with Web accessibility is a CAPTCHA. A CAPTCHA is a challenge-response test used to determine whether or not the user is a human instead of a computer or a robot. This type of element causes accessibility barriers especially to users with disabilities. This paper presents an overview about Web accessibility and CAPTCHA. Besides, an analysis of the accessibility barriers and a solution proposal depending on the type of disability is provided. Moreover, a survey of CAPTCHA approaches is introduced and its results are shown. With the knowledge gathered, a data discussion is provided. The lesson learned is that the CAPTCHA objective must be that security checks should be responsibility of websites or servers, that is, they cannot be delegated to the user.

1 INTRODUCTION

The Web sites security directive uses Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) in order to avoid input that has been generated by a computer or a bot, and authenticate that whoever is accessing is a human user. There is a trend to use this content element on many websites and sometimes it provokes accessibility barriers that avoid Web content access to people with disabilities. Besides, a lot of CAPTCHA techniques have emerged that cheat through software allowing computers or robots to access the content and that include more accessibility barriers. Users with disabilities need assistive technology to access to the Web, and CAPTCHA techniques. CAPTCHA techniques must take account it to provide a supported access to this technology. In order to deal with these accessibility barriers, most users have to be helped by other people. This fact has caused that some people protest against it as in the case of Australia (Hawkins, 2013) to ask for the elimination of Web CAPTCHA and the use of other ways of security such as SMS or mails. On the other hand, it is also important to highlight that security should be included by the server avoiding that users have to be concerned

about it.

The motivation of this paper is to carry out a study concerning accessible CAPTCHA taking into account the access features of the users with disabilities. After analysing all this information, accessibility barriers to perceive, to solve, to access (answer -typing, to pointing, ..- and submit) of the CAPTCHA for several user profiles of people with disabilities have been distinguished and proposed solutions are provide in each user profile.

The remainder of the paper is organised as follows: Section 2 covers the background of accessibility and some CAPTCHA issues. Section 4 introduces an analysis of accessibility barriers and a solution proposal. In section 5, a survey of several CAPTCHA approaches are shown. Finally, discussion of the data and lessons learned are presented in Section 7.

2 BACKGROUND

A large increase of CAPTCHA on the Web is observed. The growth of CAPTCHA is due to it is a security mechanism which can be introduced easily.. But it has not been properly carried. The accessibility has not been in mind when this element

has been integrated on the Web.

2.1 Web accessibility & CAPTCHA

It is essential to achieve an accessible Web in order to provide equal access and equal opportunity to people with diverse abilities. Therefore, an accessible CAPTCHA should be a mechanism that avoids the access of robots but not the access of a human user independently of his language, knowledge or whether the user has any kind of disability that hinders his interaction with Web content.

Currently, the use of CAPTCHA causes a great number of accessibility barriers. In order to avoid these barriers, designers have to know what the main standards and regulations have to follow in order to design an accessible CAPTCHA. Every website has to be designed according to accessibility standards that allow users to access Web content independently whether or not they have a disability.

Regarding accessibility standards is important to highlight the World Wide Web Consortium (W3C) with its Web Accessibility Initiative (WAI) (W3C, 2012). WAI includes various works such as User Agent Accessibility Guidelines (UAAG), Authoring Tool Accessibility Guidelines (ATAG) and the most important standard, the Web Content Accessibility Guidelines (WCAG) (W3C, 2008),

WCAG 2.0 is the set of accessibility guidelines most referenced in the world and since 2012 is a standard ISO/IEC 40500:2012, Information technology - W3C Web Content Accessibility Guidelines (WCAG) 2.0). Following this standard, there are several initiatives in different countries related to Web accessibility such as: Section 508, BITV 2.0, RGAA, AODA among others. These initiatives are similar to WCAG 2.0, being in some case a copy of it.

Following the WCAG 2.0, the Success Criterion 1.1.1 indicates that *Non-text content explicitly excludes the requirement to supply a text alternative to a CAPTCHA image for this very reason. But that doesn't let the developer off the hook, because it required that: ...text alternatives that identify and describe the purpose of the non-text content are provided, and alternative forms of CAPTCHA using output modes for different types of sensory perception are provided to accommodate different disabilities* (W3C, 2008).

The difficulty in making a CAPTCHA image accessible is that providing a text alternative of the image, as required for screen reader users to understand the content, also supplies the answer to

the bot. Besides, designers should take into account some techniques such as:

- Technique G143 (*Providing a alternative text that describes the purpose of the CAPTCHA*): the purpose of the technique is to provide information via the alternative text that identifies the non-text content as a CAPTCHA.
- Technique G144 (*Ensuring that the Web Page contains another CAPTCHA serving the same purpose using a different modality*): the purpose of the technique is to reduce occasions in which a user with a disability cannot complete a CAPTCHA task.

Others international standards found related to accessibility are: the ISO 9241-151, *Ergonomics of human-system interaction - Guidance on World Wide Web user interface*, the ISO 9241-171, *Ergonomics of human-system interaction - Guidance on software accessibility* that provide guidance on software accessibility and the ISO/IEC TR 29138-1, *Information technology - Accessibility considerations for people with disabilities - User needs summary*.

2.2 Disabilities

A great amount of web applications and websites present accessibility problems and they are not adapted to people taking into account their type of disability. The main types of disabilities are: Visual disability (blindness, low vision and color-blindness), Auditory disability (deaf and hearing loss), Motor disability (muscular dystrophy, multiple sclerosis, etc.) and Cognitive disability (WebAIM, 1999).

Besides this type of disability, it is important to take into consideration the degrees of a disability. For example, there are four degrees of hearing loss, mild, moderate, severe and profound.

On the other hand, a user can have more than one disability or multiple disabilities. For example, many people develop age-related impairments. These combinations of disabilities cause users have to face more accessibility barriers.

People with disabilities sometimes use other technology (software and hardware), called Assistive Technology (AT), to interact with the Web. There are types of AT that allow users to access web content. For example, for visual disability, screen readers, adapted keyboard, screen magnifiers and braille embossers can be highlighted.

The review carried out in this paper may be helpful to designers so that they can keep in mind several aspects regarding type of disabilities that

have to be taken into account when they are accomplished the design, such as: the variety of types and degrees of disabilities and the technology that can help them to allow users to access Web content.

2.3 CAPTCHA

The aim of CAPTCHA is to avoid robots and computers can register in a forum, create an email account or access to a public service. This element is also used as security mechanism when a user introduces three times a wrong password trying to access a website. On the other hand, Google uses a CAPTCHA called reCAPTCHA (Google, 2013) in order to improve book digitalization and stop spam at the same time.

The most common form of CAPTCHA is illustrated below, where distorted alphanumeric characters are presented as an image. The user is expected to type the characters they see into a form field (see Figure 1). The assumption is that most bots are not capable of recognising or interpreting the distorted alphanumeric characters and will fail the CAPTCHA.



Figure 1: Example of the most common CAPTCHA.

3 RELATED WORK

According to (W3C, 2005), there are six possible solutions using CAPTCHA or other alternative techniques, although these solutions do not solve completely the problem of the accessibility:

- Logic puzzles: uses simple mathematical word puzzles, trivia, etc. Among problems, it can be highlighted the problems that appear when users have cognitive disabilities. Other problem is that systems have to maintain a great number of questions.
- Sound output: offers a non-textual method of using the same content. The problem is that sometimes the audio can be unintelligible because of background noise and also the language can be a barrier because of different pronunciation.

- Limited-use accounts: creates limits for new users can mean of making sites unattractive targets to robots. Having to take a trial-and-error approach to determine a useful technique is a downside.
- Non-interactive checks: non-interactive mechanism can be used instead of CAPTCHA or interactive approaches. Spam filtering and heuristic checks are two non-interactive approaches.
- Federate identity systems: tries to set an identity of each client and maintain it across all sites that use the same service. This solution presents three approaches: single sign-on, public-key infrastructure solutions and biometrics.
- Other approaches: one approach is the use of artefacts of identity such as credit cards or national ID. Other approach uses SMS or email to verify the identity. The problem of using SMS is that users need a mobile phone and the use of mobile phones can cause problems for users with disabilities.

In (Holman et al., 2007), a proposal and a development of a new form of CAPTCHA that combines visual and audio information to facilitate the access of users with visual impairments is presented. Other work shows a study in which 150 on-line forums are analysed to know if they use CAPTCHA and what type of CAPTCHA is used. After the study, they concluded that the most used CAPTCHA is the text-based CAPTCHA and they realised that accessibility alternatives were rarely provided (Kuzma et al., 2011). (Shirali-Shahreza and Shirali-Shahreza, 2011) evaluates how easy CAPTCHA is for humans as well as review accessibility of the different kinds of CAPTCHA especially for visual impaired and elderly people. A new audio CAPTCHA development (called SoundsRight CAPTCHA) and an evaluation of it carried out by blind users are described (Lazar et al., 2012). In (Bigham and Cavender, 2009) a study with blind users is carried out. The study demonstrated that existing audio CAPTCHA are inadequate alternatives. Due to this fact, an optimization of the interface to solve these CAPTCHAs for non-visual use by localizing the playback controls into the answer box is presented. (Markkola, and Lindqvist, 2008) in which efforts on designing accessible voice CAPTCHA for Internet Telephony are discussed. A set of current CAPTCHAs are shown in (Roshanbin and Miller, 2013), attacks against them and an investigation about its robustness and usability are presented as well as a set of ideas to develop a CAPTCHA. Analysis of "User with disability -

CAPTCHA - Interaction".

In order to make a thorough study of the barriers that people with disabilities face when interacting with CAPTCHA, an analysis using the Scenario Method has been carried out in this work (Carroll, 1994). Scenarios are useful to get used to problems and solutions that users have to face up.

Table 1 shows a summary of the defined scenarios taking into account the different kinds of disabilities, combinations between them, their access needs and the types of barriers in the interaction with the CAPTCHA (perceive, solve and access (answer

and submit)). Besides, as a conclusion of the analysis of each scenario, proposed solutions based on standards and expert heuristics are presented. In this analysis has been considered the CAPTCHA most common, i.e., with distorted text which is presented as an image (see Figure 1).

In order to illustrate the scenarios, one of them, the scenario regarding visual disability is going to be explained. As far as visual disability is concerned, it is defined a scenario where a blind or low vision user needs AT (such as screen readers) to access information provided by a CAPTCHA. As a

Table 1: Scenarios defined according to types of disability.

| Type of disability | Acc. Barriers [Perceive, Solve, Access (Answer/ Submit)] | | | How to access CAPTCHA? | Proposed Solution |
|-----------------------------------|---|---|-------|---|--|
| Visual disability | X | | X (*) | User may need AT such as magnifier or screen reader | Provide the user the perception of the CAPTCHA with auditory modality. In addition, the WCAG 2.0 must be achieved. So, user can access with his/her AT such as keyboard to ensure that user accesses (Answer / Submit) the form. |
| Auditory disability | | | | | User should not have any problem to perceive, solve and access (answer, submit) the form because it is perceived by visual modality. In addition, the WCAG 2.0 must be achieved. |
| Motor disability | | | X (*) | User accesses with keyboard or may need AT | The WCAG 2.0 must be achieved. So, user can access with his/her AT such as with keyboard to ensure that user accesses (Answer / Submit) the form. |
| Cognitive disability | | X | X (*) | User can have difficulties to understand and solve CAPTCHA | Limiting the degree of difficulty of CAPTCHA to ensure that user solves it. In addition, the WCAG 2.0 must be achieved. |
| Visual and auditory disability | X | | X (*) | User may need AT such as Braille Displays | The WCAG 2.0 must be achieved. So, the user can use his/her AT that allows him/her to perceive and access (answer and submit) the form. |
| Visual and motor disability | X | | X (*) | User may need AT such as screen reader, magnifier, only keyboard and other AT | Provide the user the perception of the CAPTCHA with auditory modality. In addition, the WCAG 2.0 must be achieved. So, user can access with his/her AT to ensure that user accesses (Answer / Submit) the form. |
| Visual and cognitive disability | X | X | X (*) | User may need to use AT such as screen reader, magnifier and he can have difficulties to understand and solve CAPTCHA | Provide the user the perception of the CAPTCHA with auditory modality. Besides, the difficulty of how to solve the CAPTCHA should be restricted to ensure that user can solve it. In addition, the WCAG 2.0 must be achieved. So, user can access with his/her AT to ensure that he/she accesses (Answer / Submit) the form. |
| Auditory and motor disability | | | X (*) | User accesses with keyboard or may need AT | User should not have any problem to perceive and solve the CAPTCHA because it is perceived by visual modality. But, it is necessary that WCAG 2.0 is complied to ensure the access. So, user can access with his/her AT such as keyboard to ensure that user accesses (Answer / Submit) the form. |
| Auditory and cognitive disability | | X | X (*) | User can have difficulties to understand and solve CAPTCHA | User should not have any problem to perceive and solve the CAPTCHA because it is perceived by visual modality. But, it is necessary limit the degree of difficulty of CAPTCHA to ensure that user can solve it. In addition, the WCAG 2.0 must be achieved. |
| Motor and cognitive disability | | X | X (*) | User accesses with keyboard or may need AT and he/she can have difficulties to understand and solve CAPTCHA | The difficulty of how to solve the CAPTCHA should be restricted to ensure that user solves it. The WCAG 2.0 must be achieved. So, user can access his/her AT such as with keyboard to ensure that user accesses (Answer / Submit) the form. |

(*) If web content does not fulfil the WCAG 2.0.

proposed solution, incorporating an alternative audio in the CAPTCHA is proposed, which can be accessed and controlled by keyboard following the WCAG 2.0. It is essential to highlight that in this case solving a CAPTCHA is not the problem; truly, the problem is the perception of the CAPTCHA. If a user can perceive a CAPTCHA, usually he can solve it easily; being crucial that CAPTCHA and the Web page have been developed following WCAG 2.0 and provide access through AT.

In conclusion, regarding to the perception of CAPTCHA, the most affected groups of disabilities are visual and multiple disability that include visual and motor disabilities. Although incorporating audio can solve the main barriers to accessibility.

The cognitive disability and multiple disability that include cognitive disability are the groups which have observed more barriers to solve the CAPTCHA test. Although it seems clear how to avoid barriers limiting the test difficulty, in fact it is not, because the simplicity of the test may not prevent the access of malicious web robots.

With regard to access to the CAPTCHA (typing the characters into a form field and submit the form), the disability groups most affected are motor impaired users using AT, visually impaired users to access for screen reader and multiple disability that include visual and motor disabilities. In this case, the accessibility barriers are solved when the web page and the CAPTCHA comply with WCAG 2.0 that provides support to the AT.

The consequence of this analysis is that the users with more accessibility barriers and also more difficult to provide them an accessible CAPTCHA includes people who: are blind and vision impaired and cognitive disabilities (dyslexic, with difficulty reading...).

In the case of multiple disabilities, the proposed solutions are not easy to carry out, due to the existence of a conflict between the specific solutions with the aim of providing a universal solution.

4 CAPTCHA APPROACHES IN ACCESSIBILITY SCOPE

Some CAPTCHA approaches developed with accessibility requirements have been found and analysed as follows.

4.1 Survey of Accessible CAPTCHAs

For the analysis conducted, we have assumed that the web page complies with WCAG 2.0 in order to

isolate the accessibility of CAPTCHA itself and the context of use.

The approaches found are described and discussed below. Table 2 shows a summary of the analysis results obtained.

Table 2: Summary of CAPTCHA survey.

| CAPTCHA Approaches | Disabilities | | | |
|---------------------------------------|--------------|----------|-------|-----------|
| | Visual | Auditory | Motor | Cognitive |
| 1: Form test with simple question (1) | ✓ (*) | ✓ | ✓ | X |
| 2: Form test with simple (2) | ✓ (*) | ✓ | ✓ | X |
| 3: Empathy to solve | ✓ | ✓ | ✓ | X |
| 4: Advertisement to solve | X | X | ✓ | ✓ |
| 5: Recognising to an animal | X | ✓ | ✓ | X |
| 6: Access to the video | X | ✓ | ✓ | ✓ |
| 7: 3D object recognition | X | ✓ | ✓ | X |
| 8: Composing a phrase | ✓ | ✓ | X | ✓ |
| 9: Solving a Mini game | X | ✓ | X | X |
| 10: Moving the sliders | ✓ (*) | ✓ | ✓ (*) | ✓ |
| 11: HoneyPot (without using CAPTCHA) | ✓ | ✓ | ✓ | ✓ |

(*) High interdependence with the- WCAG 2.0 Compliance- and support with keyboard and AT (screen reader).

- Approach 1: a form test which presents a simple question. This question can be read by a screen reader to help blind users and enlarged by a screen magnifier to help low vision users. As downsides, it can present problems related to cognitive disabilities and it only uses Spanish language, therefore, a foreign person cannot solve this CAPTCHA.
- Approach 2: A CAPTCHA used by Aragon Government of Spain to set an appointment with the doctor. In order to set an appointment, the user, besides setting his/her National Insurance Number and surname, has to solve a CAPTCHA. To solve it, the user has to write a word that appears in red color or underlining in a sentence. Blind users can use screen magnifiers and screen readers to perceive, solve and access the CAPTCHA, although, it can present problems with users with cognitive disabilities and color-blindness considering that they cannot distinguish the color that it uses to select the word. Other disadvantage is the CAPTCHA language.

- Approach 3 (see Figure 2): this approach is based on the empathy to solve CAPTCHA, because depending on the user answer it is assumed if the user is a human or a robot. User with cognitive disabilities can have difficulties because he/she does not know what option selects. Besides, there is a language barrier considering that it is only provided in English.

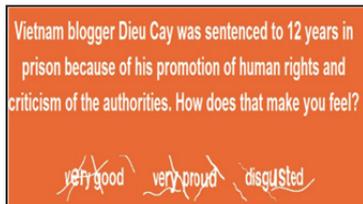


Figure 2: CAPTCHA of approach 3 (<http://captcha.civilrightsdefenders.org/>).

- Approach 4 (see Figure 3): this approach uses advertisement together with CAPTCHA. This fact allows website owner to earn money when CAPTCHA is correctly solved. As far as accessibility is concerned, this approach provokes accessibility barriers, for example, a user with visual disability does not solve CAPTCHA although the user is listening to the advertisement because the solution of the CAPTCHA appears in the image that is shown. Besides, a deaf user could not solve it, because sometimes the solution is a slogan that is listened in the video.



Figure 3: CAPTCHA of approach 4 (<http://www.solvemedia.com/>).

- Approach 5 (see Figure 4): the Animal CAPTCHA enterprise uses CAPTCHA to recognise the animals that appear in the distorted image. Users with visual disability cannot solve this type of CAPTCHA and sometimes, users with cognitive disability find problems to solve it.



Figure 4: CAPTCHA of approach 5 (<http://www.teoriza.com/captcha/example.php>).

- Approach 6 (see Figure 5): The CAPTCHA is a video in which characters are provided through an image and/or by auditory modality. On one hand, the audio may be intelligible. On the other hand, the user with visual disability can not access information of the distorted alphanumeric characters included into image; therefore, users with visual and auditory disability could have problems to perceive CAPTCHA.



Figure 5: CAPTCHA of approach 6 (<http://www.nucaptcha.com/features>).

- Approach 7 (Figure 6): the solution of Yuniti is based on 3D object recognition. As aforementioned in other solutions, users with visual disability cannot access the information and users with cognitive disability have difficulties to interpret the object if it is seen from different angles.

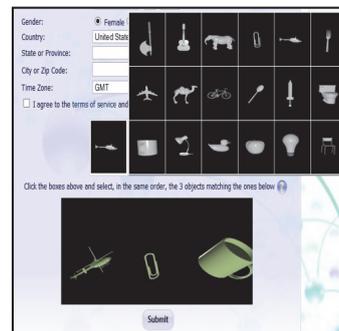


Figure 6: CAPTCHA of approach 7 (<http://www.es.yuniti.com/register.php>).

- Approach 8 (see Figure 7): this CAPTCHA shows a table with several columns which are composed of words. In order to solve CAPTCHA, user has to set a phrase selecting a word of each column. Among drawbacks, it can be highlighted the language of the CAPTCHA, in this case English, and problems for users with motor disabilities if words are so close complicating their selection via keyboard.

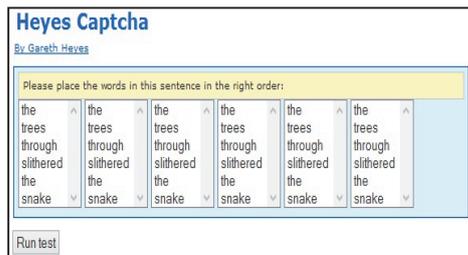


Figure 7: CAPTCHA of approach 8 (http://www.businessinfo.co.uk/labs/HeyesCaptcha3/heyecaptcha_test.php).

- Approach 9 (see Figure 8): instead of using a CAPTCHA, it is used a mini game. This CAPTCHA provokes accessibility barriers for users with visual disability, users with cognitive disability because they are not able to understand the game and users with motor disability. Although, this type of CAPTCHA provides audio, this audio can be incomprehensible. Other drawback is the language; currently it is only available in English.



Figure 8: CAPTCHA of approach 9 (<http://areyouahuman.com/>).

- Approach 10 (see Figure 9): it tries to move a slider from left to the right. It causes problems for blind users if the web content is not accessible by keyboard and screen reader and users with motor disabilities with dexterity problems. A blind user would need that his/her assistive technology allows user to know the position of CAPTCHA and to where move it.



Figure 9: CAPTCHA of approach 10 (<http://theymakeapps.com/users/add>).

Others solutions (Approach 11) which can solve the problem of CAPTCHAs are to avoid the use of them. It is considered that server should face up spam instead of user. An example is the project HoneyPot (HoneyPot, 2004). This proposal is based on that robot only interprets HTML code of web page, but they do not pay attention to CSS code, considering that, a field that user do not see could be hidden and, therefore, it could stay empty when the form is filled in. On the other hand, the robot could see the field and fill in it. In this way, a robot could be discovered. This idea avoids user to have to solve challenges that many times provoke accessibility barriers.

4.2 Discussion

The conclusion obtained is that users with cognitive disabilities are the users who have more difficulties followed by blind users. The reason of this conclusion is due to the main problem of users with cognitive disability: they do not have a good perception of the CAPTCHA. On the other hand, blind users also have problems because most CAPTCHAs are perceived through a visual canal.

Despite that the CAPTCHAs try to be accessible for people with disabilities, they do not achieve this goal completely, considering that if they provide a good solution, this solution could be easy to tackle by the robots and computers. Therefore, after the review carried out in this section, we consider the best solution is to avoid accessibility barriers by using other system to control spam instead of using CAPTCHA.

5 LEARNED LESSONS AND CONCLUSIONS

The use of CAPTCHA on the Web provokes several accessibility problems, especially for people with disabilities. This fact has motivated this work.

This paper introduces a research work which includes: a study of Web accessibility and CAPTCHA, a study of the kinds of disabilities and their accessibility barriers. According to findings of this study, the disability groups most affected by the accessibility barriers when they interact with Web content CAPTCHA are the users with cognitive and visual impairments, or multiple disability that include them. Besides, a survey and analysis of current CAPTCHA approaches in scope accessibility has been shown.

Considering that not all users can perceive, solve and access (answer y submit) the CAPTCHA, the challenge would be to design a CAPTCHA such that several alternatives to perceive the CAPTCHA and several methods to communicate the answer will be provided to the user following WCAG 2.0 techniques. In order to provide a solution proposal, as alternatives to perceive the CAPTCHA, there are two possible solutions: visual CAPTCHA and auditory CAPTCHA. But this proposal should take into account cognitive barriers.

To conclude, it is possible to design proposals CAPTCHA that can present a high level of accessibility, but unfortunately accessibility barriers continue to occur.

This lack of solutions leads us to ask ourselves whether the server has to be in charge of security without involving the final user or not. It should continue working on security solutions that prevent the use of the CAPTCHA. Some solutions already exist and can be used as using a system to control spam such as Approach 11 or email instead of using a CAPTCHA.

ACKNOWLEDGEMENTS

This work was supported by the Regional Government of Madrid under the Research Network MA2VICMR [S2009/TIC-1542], by the Spanish Ministry of Education under the project MULTIMEDICA [TIN2010-20644-C03-01] and by the European Commission Seventh Framework Programme under the project TrendMiner (EU FP7-ICT 287863).

REFERENCES

- Bigham, J., P., Cavender, A. C., 2009, Evaluating existing audio CAPTCHAs and an interface optimized for non-visual use. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'09)*, pp. 1829-1838.
- Carroll, J. M., 1994, Making use: a design representation. *Magazine Communications of the ACM*, Volume 37, Issue 12, pp. 28-35.
- Google, 2013, reCAPTCHA, <http://www.google.com/recaptcha/>.
- Hawkins, W. 2013, <http://www.change.org/en-AU/petitions/it-s-time-to-finally-kill-captcha-2>.
- Holman, J., Lazar, J., Feng, J. H., D'Arcy, J., 2007, Developing usable CAPTCHAs for blind users. *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility (Assets '07)*, pp. 245-246.
- HoneyPot, 2004, Project HoneyPot, <https://www.projecthoneypot.org/>.
- Kuzma, J., Barnes, S., Oestreicher, K., 2011, CAPTCHA accessibility study of online forums. *International Journal of Web Based Communities*, Volume 7, Number 3/2011.
- Lazar, J., Feng, J., Brooks, T., Melamed, G., Wentz, B., Holman, J., Olalere, A., Ekedebe, N., 2012, The SoundsRight CAPTCHA: an improved approach to audio human interaction proofs for blind users. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'12)*, pp. 2267-2276.
- Markkola, A., Lindqvist, J., 2008, Accessible voice CAPTiCHA for Internet Telephony. *Proceedings of the 2008 Symposium on Accessible Privacy and Security*.
- Roshanbin, N., Miller, J., 2013. A survey and analysis of current CAPTCHA approaches. *Journal of Web Engineering*, Vol. 12, No. 1&2, pp. 1-40.
- Shirali-Shahreza, S., Shirali-Shahreza, M.H., 2011, Accessibility of CAPTCHA methods. *Proceedings of the 4th ACM workshop on Security and artificial intelligence (AISec'11)*, pp. 109-110.
- WebAIM, Web Accessibility in Mind, 1999, <http://webaim.org/articles/>.
- W3C, 2005, Inaccessibility of CAPTCHA - Alternatives to Visual Turing Tests on the Web, <http://www.w3.org/TR/turingtest/#solutions>.
- W3C, 2012, Web Accessibility Initiative (WAI), <http://www.w3.org/WAI/intro/components.php>.
- W3C, WAI, 2010c, Web Content Accessibility Guidelines (WCAG), <http://www.w3.org/WAI/intro/wcag.php>.
- W3C, 2008, Web Content Accessibility Guidelines (WCAG) 2.0, <http://www.w3.org/TR/WCAG20/>.

Fuzzy-Ontology-Enrichment-based Framework for Semantic Search

Hajer Baazaoui-Zghal and Henda Ben Ghezala

Riadi-GDL Laboratory, ENSI, Manouba University, Tunis, Tunisia

{hajer.baazaouizghal, henda.bg@riadi.rnu.tn}

Keywords: Fuzzy Ontology, Web Search, Query Reformulation.

Abstract: The dominance of information retrieval on the Web makes integrating and designing ontologies for the on-line Information Retrieval Systems (IRS) an attractive research area. In addition to domain ontology, some attempts have been recently made to integrate fuzzy set theory with ontology, to provide a solution to vague and uncertain information. This paper presents a framework for semantic search based on ontology enrichment and fuzziness (FuzzOntoEnrichIR). FuzzOntoEnrichIR main components are: (1) a fuzzy information retrieval component, (2) an incremental ontology enrichment component and (3) an ontology repository component. The framework aims on the one hand to capitalize and formulate extraction-ontology rules based on a meta-ontology. On the other hand, it aims to integrate the domain ontology enrichment and the fuzzy ontology building in the IR process. The framework has been implemented and experimented to demonstrate the effectiveness and validity of the proposal.

1 INTRODUCTION

Ontologies are defined as an explicit formal specification of a shared conceptualization. They can be classified as lightweight ontologies gathering concepts and relations' hierarchies which can be enriched by classical properties called axiom schemata (algebraic properties and signatures of relations, abstraction of concepts, etc.) and heavyweight ontologies which add properties to the semantics of the conceptual primitives and are only expressible in axiom domain form. The axioms schemata describe the classical properties of concepts and relations (subsumption, disjunction of concepts, algebraic properties and cardinalities of the relations, etc.). The domain axioms characterize domain properties expressible only in an axiom form. They specify the formal semantics constraining the conceptual primitive interpretation.

Currently, ontologies are playing a fundamental role in knowledge management and semantic Web. Building ontology manually is a long and tedious task. Thus, many approaches have been proposed during the last decade to make this task easier. Information Retrieval (IR) deals with models and systems aiming to facilitate accessibility to sets of documents and provide to users the corresponding ones to their needs, by using queries. Generally, Information Retrieval System (IRS) integrates techniques allowing selection of relevant information. The first research on ontologies for the IRS dates back to the late 90s

(McGuinness, 1998), and aims to remedy limits of the traditional IRS based on the keywords. This research topic presents one of the main actual axes of the semantic Web.

In addition to domain ontology, the integration of the fuzzy logic shows that it presents an interesting way to solve uncertain information problems. In fact, fuzzy logic is used in IR to solve the ambiguity issues by defining flexible queries or fuzzy indexes (e.g., (Baazaoui-Zghal et al., 2008)). A fuzzy ontology can be considered as an extension of domain ontology by embedding a set of membership degrees in each concept of the domain ontology and adding fuzzy relationships among these fuzzy concepts (Zhou et al., 2006).

In this paper, we present a framework for semantic search based on fuzzy ontologies. It includes an ontology repository (meta-ontology generating a domain ontology and ontology of domain services), incremental approach of domain ontology learning and fuzzy ontology enrichment method. The framework has been implemented to demonstrate the proposal effectiveness and to evaluate it.

The remaining of this paper is organized as follows. Section 2 presents related works to information retrieval and fuzzy ontologies. Section 3 describes our proposal. Section 4 presents and discusses some experimental results of our framework. Finally, section 5 concludes and discusses directions for future research.

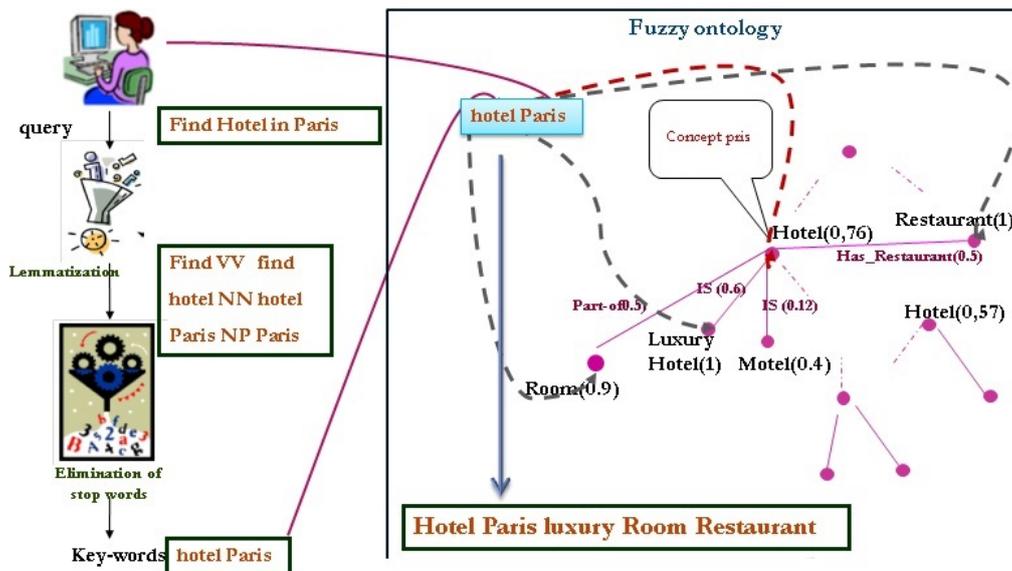


Figure 1: An example of a fuzzy ontology.

2 RELATED WORKS AND MOTIVATIONS

Several studies were presented showing how the fuzzy logic could be integrated to IRS, to solve uncertain information problems (Zhou et al., 2006). We precise here, that in a fuzzy ontology, each index term or object is related to every term (or object) in the ontology, with a degree of membership assigned to the relationship and based on fuzzy logic.

The fuzzy membership value μ is used for the relationship between terms or objects, where $0 < \mu < 1$, and μ corresponds to a fuzzy membership relation such as "strongly", "partially", "somewhat", "slightly"

etc, where for each term: $\sum_{i=1}^{i=n} \mu_i = 1$;

n is the number of relations that a particular object has, $n = (N - 1)$, with N representing the total number of objects in the ontology (Lee et al., 2005).

The insertion of the fuzzy logic and the ontology in the process of information retrieval has improved the quality and the precision of the returned results. Thus, integration of the fuzzy ontology into the IR process is an interesting area of research and can lead to more relevant results than in the case where ontology and fuzzy logic are used separately (Chien et al., 2010; Bordogna et al., 2009; Calegari and Ciucci, 2006). Several existing IRSs (Zhou et al., 2006; Chien et al., 2010; Calegari and Ciucci, 2006) generally use semi-automatic or automatic methods, which allow the fuzzification only of the "IS-A" rela-

tion. In addition, from the state of the art (Chien et al., 2010; Colleoni et al., 2009), it is noticed that there is a lack of information retrieval system integrating fuzzy ontology allowing a document classification and assisting users in their searches.

First, in (Sayed et al., 2007), document classification is not based on fuzzy ontologies.

In fact, classification based only on domain ontology could not take into account dynamic aspect of fuzzy ontology, mainly when the aim is to improve query reformulation and information retrieval results. Both in (Parry, 2006; Akinribido et al., 2011), only "IS-A" relations are taken into account. Nevertheless, all relations are important mainly in case of query reformulation.

From the conducted survey made on methods for fuzzy ontology construction, we have noticed that automatic methods can take as input a database (Lee et al., ; Quan et al., 2006), a documentary corpus (Widyantoro and Yen, 2001) or an existing ontology (fuzzification) (Parry, 2006),(Sayed et al., 2007), (Chien et al., 2010), (Calegari and Ciucci, 2006).

Figure 1 illustrates an example of a fuzzy ontology. The number related to relations represents the membership value of the relationship between the concept "Hotel" and other concepts (room, Suite). The related value to a concept describes the importance of this concept into the ontology. These different relations and concepts will have different membership values depending on the context of the query, and particularly the user's view of the world.

In this paper, our main objective is to develop a

framework allowing ontology's building for the semantic Web. The proposed framework includes an ontology repository (meta-ontology generating a domain ontology and ontology of domain services, and fuzzy ontology), incremental approach of domain ontology learning, fuzzy ontology building method and ontology based retrieval process.

So, the aims of this paper can be summarized in:

- The integration of the domain ontology enrichment and the fuzzy ontology building in the IR process.
- The capitalization and formulation of extraction ontology rules based on a meta-ontology aiming to explicitly specify knowledge about the concepts, relationships, instances and axioms extraction, the learned patterns and frames, and the semantic distance

3 FUZZY-ONTOLOGY-ENRICHMENT-BASED FRAMEWORK FOR SEMANTIC SEARCH

The general structure of the framework (FuzzyOntoEnrichIR) is given by Figure 2. FuzzOntoEnrichIR framework is based on a fuzzy ontology building, an incremental ontology learning and an ontology repository.

The main FuzzOntoEnrichIR components are: a fuzzy ontology building component, an incremental ontology learning component and an ontological repository component.

The first component is composed of:

- Two methods of information retrieval based on a domain ontology and an individual fuzzy ontology.
- An automatic method of fuzzy ontology building: allowing the fuzzification of all existing relations in the initial domain ontology initial, and assuring the updating of membership values at the end of every information retrieval, which is made dynamically by the user
- A classification of documents by service using a domain ontology

This first component aims to automate the collection of the relevant documents which will be used as entry of the second component

The second component is based on a meta-ontology (which is a high-level ontology of abstraction (Baazaoui-Zghal et al., 2007a)) and an incremental ontology learning which may require enrichment

phase. It allows incremental construction of ontologies from the Web documents. Thus, this component proposes a composite ontological architecture of three interdependent ontologies: a generic ontology of web sites structures, a domain ontology and a service ontology.

These offer a representation of the domain and services behind Web content, which could be exploited by the semantic search engines. This latter is instantiated with the contextual information of concepts and relations of the ontology extracted incrementally from texts. The semi-automatic construction of the domain ontology is the main objective of this component.

The third component is composed of a meta-ontology generating a domain ontology and ontology of domain services, and fuzzy ontology.

The details related to each component will be given in the next subsections.

3.1 Fuzzy Ontology Building Component

FuzzyOntoEnrichIR integrates an automatic fuzzy ontology building method, with an automatic fuzzification of all the existing relations in the domain ontology, not restricted to "Is-a" relations. Indeed, in conventional ontologies, particular objects may occur in multiple locations. So, a simple expansion that does not understand the intended location of the query term may lead to many irrelevant results being returned. A fuzzy ontology membership value could therefore be used to identify the most likely location in the ontology of a particular term. Each user would have own values for the membership assigned to terms in the ontology, reflecting their likely information need and worldview.

Then, the use of an individual fuzzy ontology approach allows the convenient representation of the relationships in a domain according to a particular view, without sacrificing commonality with other views; the ontology framework is common, just the membership values are different. An individual fuzzy ontology using an automatic fuzzification based is built.

Initialization of Membership Values

We propose to build an individual fuzzy ontology using an automatic fuzzification based on Jiang-Conrath similarity measure (Jiang and Conrath, 1997). To calculate *IC* (Information Content), we use the formula presented by (Seco et al., 2004) which is based on the structure of the ontology hierarchy. In fact, this frequency has the advantage of bringing the occurrence frequency of the concept itself and the

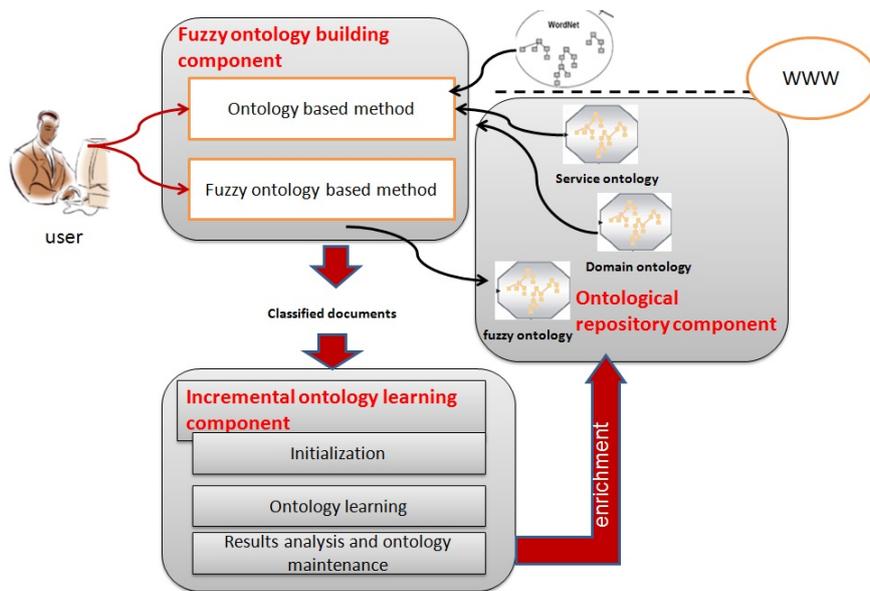


Figure 2: FuzzyOntoEnrichIR framework architecture.

concepts it subsumes, which allows supporting all relations' types.

Updating the Membership Value of Concepts and Relations

We suppose that a defined fuzzy ontology is not available in any context. Thus, it is necessary to define an update process of fuzzy values, taking into account the user's needs. The membership value should consider the previous values, the retrieved documents and the query. In the literature, there are researchers that have presented similar ways of updating membership value (Calegari and Ciucci, 2006; Parry, 2006). Inspired by these methods, we have integrated in our case two updating membership values respectively for concepts and relations.

$$\mu_{new} = \mu_{old} + \frac{\mu - \mu_{old}}{Q + 1} \quad (1)$$

where μ_{old} is the current membership value, Q is the number of update performed to this value and μ_{new} is the new value. μ is a value that evaluates the new change added to the relation or the concept. μ must take into account the query and the returned documents content that have been selected by the user.

The fuzzy ontology is used for query reformulation and for documents and query indexing. A fuzzy ontology is an individual an ontology owned by each user.

To show the purpose of the given formulas, we take as example the query sent by the user containing the concept "Hotel". The framework computes

themaasures of this concept and all its related concepts (like: Rate, Restaurant, Motel...) using the returned documents selected by the user. Finally, the membership value of relations using the formula 5 are also updated. In this example the membership value of the relation "has-restaurant" between "Hotel" and "restaurant" will be updated (cf. Figure 3).

3.2 Ontological Repository Component

A dedicated architecture is proposed based on two interdependent ontologies to build a knowledge-base of a particular domain, constituted by a set of Web documents, and associated services. Thus, two ontologies are distinguished: domain ontology and ontology of domain services, which are in interaction. These two ontologies are built in accordance with the meta-ontology. Domain ontology is a set of concepts, relations and axioms that specify shared knowledge concerning a target domain. Ontology of domain services specifies for each service, its provider, its interested users, possible process of its unrolling, main activities and tasks that constitute this service (Baazaoui-Zghal et al., 2007b).

This ontology contains axioms specifying the relations between domain services and precise main domain concepts which identify each service. These ontologies are semantically linked and relationships between them are defined. The meta-ontology is a specification of meta-models of domain ontology and ontology of domain services. Besides, knowledge concerning the semi-automatic construction of domain

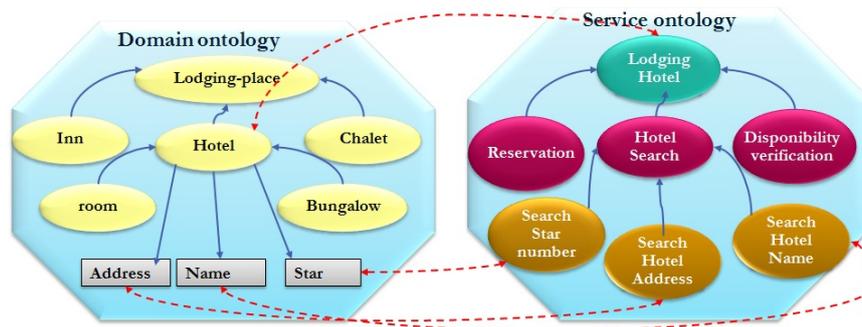


Figure 3: Relation between domain ontology and service ontology.

ontology is also specified by this meta-ontology.

The proposed architecture is composed of three ontologies, namely a generic ontology of web sites structures, domain ontology and service ontology.

The generic ontology or meta-ontology contains knowledge representation related to each ontology, which required knowledge for ontology construction, knowledge representation specified by the meta-model related to these ontologies. It is mostly based on generic concepts: "meta-concept", "meta-relation", and "meta-axiom". The class of "meta-concept" is divided into subclasses which represent respectively the domain meta-concept, the meta-concept of domain services and the meta-concept of element. Besides, the class "meta-relation" and the class "meta-axiom" are designed in the same way. The meta-ontology is, consequently, made up of three homogeneous knowledge fields. The first field is a conceptualization of knowledge related to learning concepts, relations and axioms related to a target domain.

Besides, for each instance of the class *Domain-Concept* and the class *Domain-Relation*, the technique leading to its discovery is specified.

The service ontology specifies the common services that can be solicited by web users and can be attached to several ontologies defined on subparts of the domain (*cf.* Figure 3 showing the relation between domain ontology and service ontology)

3.3 Incremental Ontology Learning Component

The incremental ontology learning component is based on a process of ontology learning from Web content according to LEO_By_LEMO (LEarning Ontology BY Learning_Meta-Ontology) approach (Baazaoui-Zghal et al., 2007b). Our approach is based on learning rules of ontology extraction from texts in order to enrich ontologies in three main

phases:

- Initialization phase
- Incremental phase of learning ontology
- Result analysis phase.

The initialization phase is dedicated to data source cleaning. The input of this phase is constituted by a minimal ontology, the meta-ontology, the terminological resource "Wordnet" (Miller", 1995) and a set of Web sites delivered by a search engine and classified by domain services. A minimal ontology is designed and built to be enriched in the second phase. It is called "minimal domain ontology" as the number of concepts and relations are reduced. Consequently, data source preparation consists of: searching Web documents related to the domain corresponding to a query based on concepts describing a target service (these concepts are obtained from the projection of the corresponding service specified in the ontology of domain services), selecting Web sites provided by a search engine tool (the number of chosen sites is limited because analyzing an important number of Web pages requires very important execution time), classifying Web pages according to domain services.

Finally, cleaning Web pages by eliminating markup elements and images, text segmentation and tagging in order to obtain a tagged textual corpus. One hypothesis is that we deal with Web documents written in a target language. The meta-ontology adjustment is thus done according to linguistic knowledge related to the target language. The second phase is a learning iterative process. Each one of the iterations is made up of two main steps. The first one is the meta-ontology instantiation and the second one enables us to apply the meta-ontology axioms related to the learning of ontology. An iteration is processed in two steps. In the first step, techniques are applied to the corpus. In this context, we have adapted the construction of a word space (Baazaoui-Zghal et al., 2008) by applying the N-Gram analysis instead of a 4-

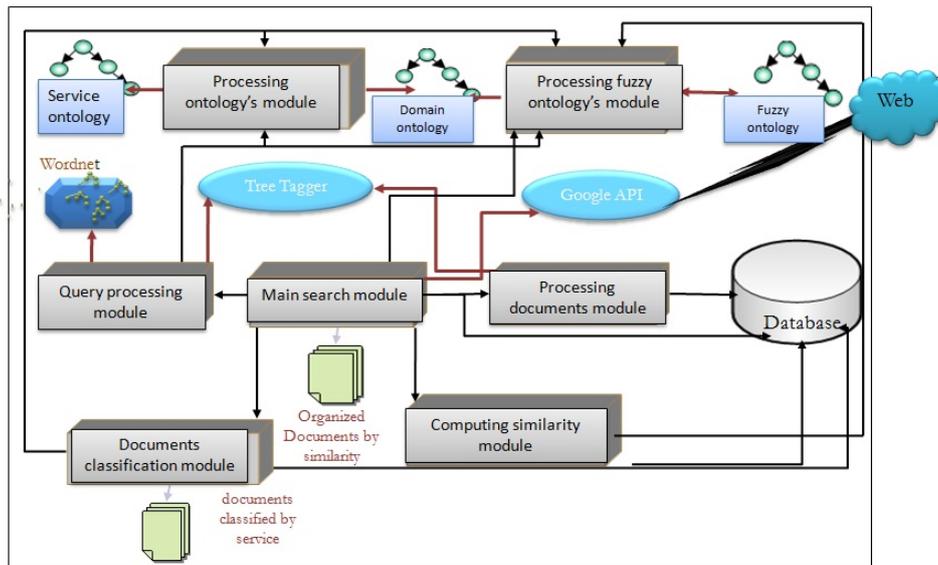


Figure 4: Modular architecture framework.

gram analysis. We have also proposed a disambiguation algorithm (Baazaoui-Zghal et al., 2007b). It aims to determine the right sense of a lexical unit. This algorithm is based on the study of term co-occurrence in the text and the selection of the adequate sense. Besides, we propose to use many similarity measures to build the similarity matrix which describes the contextual similarity between concepts.

The last phase is useful to verify the coherence of enriched ontology by analyzing learning results. We admit that the maintenance of the meta-ontology allows the readjustment of the rules according to the results obtained in order to improve the ontology construction during a further execution of the second phase of the process. Moreover the correction of meta-ontology generates a more valid ontology scheme and richer.

4 EXPERIMENTATION AND RESULTS ANALYSIS

The implementation and experimentations of the proposed framework, have been done in order to evaluate the proposed architecture. Figure 4 gives an idea about the developed modules and the application structure. FuzzEnrichIR is composed of the construction and updating module which allows manipulation of the fuzzy ontology. The processing module regroups classes assuring the different treatments done on the request and on the documents (as indexing and downloading). The class module regroups the

different useful classes to the document classification by service. A pre-processing module of data sources which pre-process textual corpus, its POS (Part-of-Speech) tagging and importation of terminological and conceptual resources (minimal ontology, Ontology of domain services and terminological resource "Wordnet"). An editing module of the meta-ontology which allows concepts and ontology axioms update by integrating the Plug-in of Protege-OWL tool. A module for domain ontology generation. An alimentation module of the meta-ontology which consists of conceptual elements in the meta-ontology from text and implements the incremental process of ontology domain construction proposed by the "LEO-By-LEMO" approach. Finally, a module of domain ontology learning which is the result of the association and the development of learning techniques set (concepts and relations).

The framework was implemented in Java, providing an online service and using the Jena Api to handle ontologies and Google Api to search through the Web.

Several experiments were conducted to investigate the performance of our proposal and to evaluate:

- The impact of ontology enrichment on information retrieval relevance
- The impact of fuzzy ontology enrichment on information retrieval relevance with and without update

The adopted protocol is centered on users, and the used data for the experimentation and the evaluation was composed of a domain ontology and users' requests. Fifteen queries in the tourism's domain and

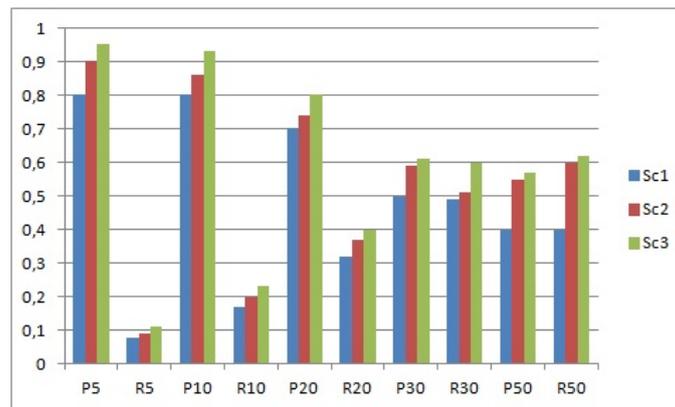


Figure 5: Scenarios' results.

ten users were considered. Users evaluate the results obtained by using the domain ontology, the individual fuzzy ontology and the updated individual fuzzy ontology.

Three scenarios were designed to evaluate the proposed framework:

- Scenario based on domain ontology (*Sc1*)
- Scenario based on individual fuzzy ontology, without update (*Sc2*)
- Scenario based individual fuzzy ontology, with 4 updates (*Sc3*)

Figure 5 shows the results in terms of precision (P) and recall (R), for Top 5, 10, 20, 30 and 50 retrieved documents. To evaluate the recall values, we consider the same queries used for the precision. We analysed the first 50 relevant URL's returned by every scenario. The obtained precision and recall related with FuzzOntoEnrichIR for scenario *Sc3* are clearly higher than the ones obtained by *Sc1* and *Sc2*. Indeed, results related to the comparison of exact precision obtained with FuzzOntoEnrichIR based on domain ontology only, FuzzOntoEnrichIR based on fuzzy ontology with and without update, present a precision of 0.95 which is superior to the two other scenarios. These results show that the use of fuzzy ontology supporting update process increases the precision more than the use of simple domain ontology.

To complete these results we computed the improvements given by Table 1, which show that individual fuzzy ontology updated four times is reported to achieve 18,75% precision and 37,50% recall. Indeed, the results show that the use of fuzzy ontology increased the precision more than the use of simple domain ontology.

In the experimentations, the initial used ontology and fuzzy ontology are composed of 8 concepts, enriched with 20 concepts after the first iteration, with

Table 1: The improvement of the average recall and the average precision of the FuzzEnrichOntoIR framework.

| Precision/ Recall | Precision improvement TS3 vs. TS1 (in %) | Recall improvement TS3 vs. TS1 (in %) |
|----------------------|---|--|
| P/R 05 | 18,75 | 37,50 |
| P/R 10 | 16,25 | 35,29 |
| P/R 20 | 14,28 | 25,00 |
| P/R 30 | 12,00 | 22,44 |
| P/R 50 | 07,50 | 17,50 |

20 concepts, enriched with 27 concepts after the second iteration, enriched with 40 concepts after the third iteration, and enriched with 100 concepts after the fourth iteration. From the obtained results after the experimentations, we note that the incremental enrichment of the domain ontology improves the relevance.

However, relevance becomes stable after the third iteration, when the size of the ontology is enough great to cover a unique complex query. Indeed, the 100 concepts don't cover the field of the same request, but they serve to other composed requests in the same domain. For this reason, the variance of the relevance of the first iteration of the enrichment has a remarkable impact on the relevance of improvement.

5 CONCLUSIONS AND FUTURE WORKS

In this paper, we proposed a fuzzy-ontology-enrichment framework based on fuzzy ontology, namely FuzzOntoEnrichIR. Since ontologies have proven their capacity to improve IR, fuzzy ontology-based IR is becoming an increasing research area. FuzzOntoEnrichIR's framework takes place in four

main phases:

- Initialization of membership values,
- Updating the membership value of concepts and relations,
- Updating the membership value of the existing concepts in the user's query
- Updating the membership value of relations related to the existing concepts in the user's query.

Fuzzy ontologies building method is integrated to IR process, and returned results are classified taking into account fuzzified relations.

So, in this work, our first contribution concerns the fuzzy ontology's building process. Our method considers automatic fuzzification of a domain ontology taking into account both taxonomic and non taxonomic relations, however, all relations are important mainly in case of query reformulation.

Second contribution concerns the integration of our fuzzy ontology method into the IR process. Indeed, query reformulation is based on the weights associated to all the relations existing in the fuzzy ontology, and this fuzzy ontology is used to classify documents by services.

Finally, the obtained results establish the great interest and FuzzOntoEnrichIR's contribution to improve the performance of the retrieval task. Experiments and evaluations have been carried out, which highlight that overall achieved improvement are obtained thanks to the integration of fuzzy ontologies into IR process, integration of update and classification. These components contribute to significantly increase the relevance of search results, by enhancing documents ranking as shown by the obtained results.

As an evolution of this work, integration of modular ontologies in order to facilitate the updates is in progress. Otherwise, the ontology will be extended to different domains so that architecture will support a multi-domain use of the ontology. A multi-domain retrieval based on modular and fuzzy ontologies will be possible.

REFERENCES

- Akinribido, C. T., Afolabi, B. S., Akhigbe, B. I., and Udo, I. J. (2011). A fuzzy-ontology based information retrieval system for relevant feedback. In *International Journal of Computer Science Issues*.
- Baazaoui-Zghal, H., Aufaure, M.-A., and Mustapha, N. B. (2007a). Extraction of ontologies from web pages: Conceptual modelling and tourism application. *Journal of Internet Technology (JIT), Special Issue on Ontology Technology and Its Applications*, 8:410–421.
- Baazaoui-Zghal, H., Aufaure, M.-A., and Mustapha, N. B. (2007b). A model-driven approach of ontological components for on-line semantic web information retrieval. *Journal of Web Engineering*, 6(4):309–336.
- Baazaoui-Zghal, H., Aufaure, M.-A., and Soussi, R. (2008). Towards an on-line semantic information retrieval system based on fuzzy ontologies. *JDIM*, 6(5):375–385.
- Bordogna, G., Pagani, M., Pasi, G., and Psaila, G. (2009). Managing uncertainty in location-based queries. *Fuzzy Sets and Systems*, 160(15):2241–2252.
- Calegari, S. and Ciucci, D. (2006). Towards a fuzzy ontology definition and a fuzzy extension of an ontology editor. In *ICEIS (Selected Papers)*, pages 147–158.
- Chien, B.-C., Hu, C.-H., and Ju, M.-Y. (2010). Ontology-based information retrieval using fuzzy concept documentation. *Cybernetics and Systems*, 41(1):4–16.
- Colleoni, F., Calegari, S., Ciucci, D., and Dominoni, M. (2009). Ocean project a prototype of aiwbases based on fuzzy ontology. In *ISDA*, pages 944–949.
- Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. *CoRR*, cmp-lg/9709008.
- Lee, C.-S., Jian, Z.-W., and Huang, L.-K. (2005). A fuzzy ontology and its application to news summarization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 35(5):859–880.
- Lee, C.-W., Shih, C.-W., Day, M.-Y., Tsai, T.-H., Jiang, T.-J., Wu, C.-W., Sung, C.-L., Chen, Y.-R., Wu, S.-H., Hsu, and Wen-Lian. Asqa: Academia sinica question answering system for ntcir-5 clqa.
- McGuinness, D. L. (1998). Ontological issues for knowledge-enhanced search. In *Proceedings of Formal Ontology in Information Systems*.
- Miller, G. A. (1995). "wordnet: A lexical database for english". *Commun. ACM*, 38(11):39–41.
- Parry, D. (2006). Chapter 2 fuzzy ontologies for information retrieval on the {WWW}. In Sanchez, E., editor, *Fuzzy Logic and the Semantic Web*, volume 1 of *Capturing Intelligence*, pages 21 – 48. Elsevier.
- Quan, T. T., Hui, S. C., Fong, A. C. M., and Cao, T. H. (2006). Automatic fuzzy ontology generation for semantic web. *IEEE Trans. Knowl. Data Eng.*, 18(6):842–856.
- Sayed, A. E., Hacid, H., and Zighed, D. A. (2007). Using semantic distance in a content-based heterogeneous information retrieval system. In *MCD*, pages 224–237.
- Seco, N., Veale, T., and Hayes, J. (2004). An intrinsic information content metric for semantic similarity in wordnet. In *ECAI*, pages 1089–1090.
- Widyantoro, D. and Yen, J. (2001). A fuzzy ontology-based abstract search engine and its user studies. In *Fuzzy Systems, 2001. The 10th IEEE International Conference on*, volume 3, pages 1291–1294.
- Zhou, L., Zhang, L., Chen, J., Xie, Q., Ding, Q., and Sun, Z. X. (2006). The application of fuzzy ontology in design management. In *IC-AI*, pages 278–282.

A Semantic-based Data Service for Oil and Gas Engineering

Lina Jia^{1,2}, Changjun Hu^{1,2}, Yang Li^{1,2}, Xin Liu^{1,2}, Xin Cheng^{1,2}, Jianjun Zhang³ and Junfeng Shi³

¹*School of Computer & Communication Engineering, University of Science & Technology Beijing,
No.30 Xueyuan Road, Haidian District, Beijing, China*

²*Beijing Key Laboratory of Knowledge Engineering for Materials Science,
No.30 Xueyuan Road, Haidian District, Beijing, China*

³*Research Institute of Exploration & Development, CNPC, No.20 Xueyuan Road, Haidian District, Beijing, China
{jialinabetter, hu.cj.mail, mailbox.liyang, ustb.liuxin, chengxin0613}@gmail.com, sjf824@yahoo.com.cn*

Keywords: Semantic-based Data Integration, Ontology, Data Service, Data of Oil and Gas Engineering.

Abstract: For complex data sources of oil and gas engineering, this paper summarizes characteristics and semantic relationships of oil data, and presents a semantic-based data service for oil and gas engineering (SDSOge). The domain semantic data model is constructed using ontology technology, and semantic-based data integration is achieved by ontology extraction, ontology mapping, query translation, and data cleaning. With the semantic-based data query and sharing service, users can directly access distributed and heterogeneous data sources through the global semantic data model. SDSOge has been used by upper applications, and the results show that SDSOge is efficient in providing a comprehensive and real-time data service, saving energy, and improving production.

1 INTRODUCTION

With continuous expansion of the scale of petroleum exploration industry, the domain of oil and gas engineering has accumulated massive data resources, like production data, geological structures, equipment data, well structure data, etc. These data are large in scales, numerous in kinds, complex in relationships and various in characteristics:

1) **Distribution:** In oil fields, different types of data are stored in different specialized databases, such as production database, geological database, and equipment database. But applications of oil and gas engineering require various data from different databases.

2) **Heterogeneity:** Each specialized database has its own data organizing and naming convention, which results in system, syntax, structure, and semantic heterogeneity. (1) **System heterogeneity:** Different data have different operating environments, such as hardware configurations and operating systems. (2) **Syntax heterogeneity:** Different data are stored in different forms in the computers. Some are in relational databases, while some are in text files. (3) **Structure heterogeneity:** Similar data are represented in different data schemas. (4) **Semantic heterogeneity:** Similar data

have different semantic understandings, or different data have the same meaning, which has traditionally been divided into homonyms and synonyms.

3) **Complex Semantic Relationships:** There are complex relationships between different data.

4) **Real-time Performance:** The data of oil and gas engineering is dynamic and instantly updated with high real-time demand.

The characteristics of data of oil and gas engineering bring unprecedented challenges for conventional data management. On the one hand, with the differences in data schemas of different oil fields and the shortage of data management and naming rules, it is necessary to shield heterogeneity of underlying data to establish a global semantic data model for the domain of oil and gas engineering, which can maintain the unification of rules and standards, and data management platform. On the other hand, applications of oil and gas engineering are typically data-intensive. Data are the source of these applications and various data from different specialized databases are needed, but databases of oil fields are highly autonomous, which makes data interacting and sharing more difficulty. Thus semantic-based data integration is urgently in need, which can provide a unified and semantic-based interface to access the underlying data sources directly and implement data sharing.

This paper presents a semantic-based data service for oil and gas engineering named SDSOge, which provides a rich semantic view of the underlying data and enables an advanced querying functionality. Users can enjoy a plug-and-play (Mezini and Lieberherr 1998) model and have direct access to the distributed and heterogeneous data resources anywhere. In addition, the data service offers a semantic reasoning functionality, which can reason implicit knowledge behind the complicated semantic relationships.

SDSOge firstly extracts local ontologies from schemas of data sources using ontology technology, and then establishes a completed global ontology which can support each local data source. Furthermore, an interface is set up to access underlying data sources, which can eliminate differences in data sources and provide a uniform and transparent semantic-based data query service. Finally, the cleaned standard data are returned to the upper applications.

This paper is organized as follows. Section 2 introduces related work while section 3 describes the architecture of SDSOge and its implementation in details. The usage of SDSOge system and its production application pointing out the advantages comparing to previously employed techniques are illustrated in section 4. Finally, the conclusion and directions for future work are given in section 5.

2 RELATED WORK

As the complexity of data brings more and more challenges, a new approach of data service is becoming increasingly necessary.

Carey et al. (2012) survey three kinds of popular data services, service-enabling data stores, integrated data services and cloud data service, respectively. But none of the three considers semantic association.

Doan et al. (2004) introduce the special issue on semantic integration. They point out that 60-80% of the resources in a data sharing project are spent on reconciling semantic heterogeneity. Halevy et al. (2005) describe successes, challenges and controversies of enterprise information integration. Kondylakis et al. (2009) review existing approaches for ontology/schema evolution and give the requirements for an ideal data integration system.

Bellatreche et al. (2006) propose the contribution of ontology-based data modeling to automatic integration of electronic catalogues within engineering databases, but this method assumes the data source itself does not have enough semantic

information.

Ghawi and Cullot (2007) propose a semantic interoperability from relational database to ontology, but it only considers the case of one data source.

In order to make a more intuitive view of mapping, many mapping tools like COG, DartGrid, VisAVis, and MAPONTO, are developed. These tools need users to build mappings in an interactive way.

Data from different domains have different characteristics. These data are the basis of scientific research in the fields. Semantic-based data integration and data services for domain-oriented ontology are hotspots of current research. Establishment of semantic data models, and integration and application of semantic data in scientific fields are important aspects worthy of discussion and research.

3 SDSOGE ARCHITECTURE AND IMPLEMENTATION

3.1 System Architecture

SDSOge provides a global semantic data model and APIs for users and upper applications to send queries and receive desired data. Service consumers need not to know the source and original schema of data. Figure 1 shows the architecture of SDSOge.

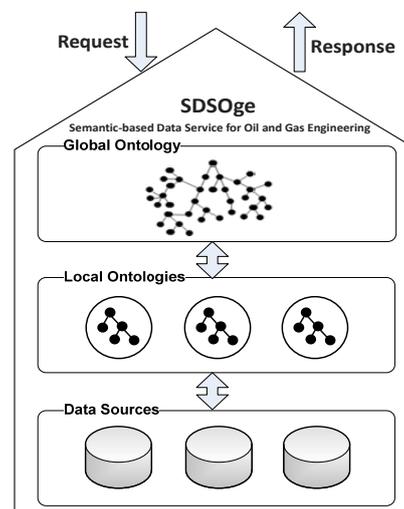


Figure 1: SDSOge Architecture.

3.2 Global Ontology Construction

There are four steps to establish the global ontology.

First of all, filter data of oil and gas engineering field and get entities that system needs and relationships between the entities. Next, extract schema information of databases to establish local ontologies using ontology technology. Then, the global ontology can be built through standardizing names of properties with the synonym table, and further refining, improving and merging of local ontologies. Finally, adding semantic constraint rules and reasoning mechanisms to form a complete and semantically rich global ontology. The global ontology construction process is shown in Figure 2.

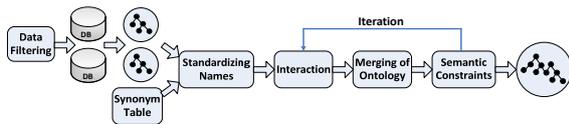


Figure 2: The global ontology construction process.

3.2.1 Data Filtering

In the field of petroleum exploration and development, data involve more than 20 professional aspects, and data of oil and gas engineering domain are just a part of them. So we should firstly define the basic scope of required data to form entities, attributes and relationships between entities referring to the data dictionary.

Take block data entity and sucker rod data entity as examples, the corresponding entity models are as follows.

Block data entity:

$E(\text{BlockInfo}) = \{\text{block_name, oil_density, permeability, reservoir_depth, \dots}\}$

Sucker rod data entity:

$E(\text{SuckerRodInfo}) = \{\text{sucker_rod_id, diameter, length, \dots}\}$

3.2.2 From Relational Database to Local Ontology

Based on the features of tables and constraints between tables in the specialized databases, rules from relational database to local ontology are defined as follows.

Rule1: Convert each table T into a class or a subclass C_T (OWL: Class or OWL: Subclass).

Rule2: Convert C_{T_j} into a subclass of C_{T_i} , if the foreign key of table T_i corresponds to the primary key of table T_j (OWL: Subclass).

Rule3: Convert the foreign key of table T into object property OP_T (OWL: ObjectProperty).

Rule4: Convert the primary key of table T into the datatype property with functional property DP_T (OWL: DatatypeProperty).

Rule5: Convert other columns of table T into

data properties DP_T (OWL: DatatypeProperty).

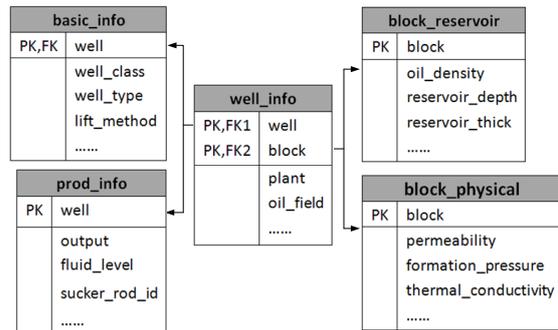


Figure 3: Tables in production database (partial).

Figure 3 shows the schema of a few tables in production database. According to the mapping rules above, the local ontology can be generated automatically. The relationships between classes are foreign key constraints in the database, as shown in Figure 4.

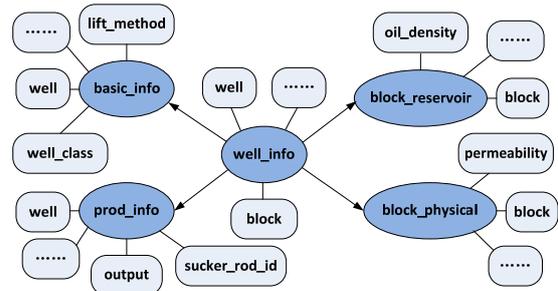


Figure 4: Local ontology of production database (partial classes).

3.2.3 From Local Ontologies to Global Ontology

The process of local ontologies to global ontology is divided into three steps, renaming of properties, merging of classes, and combination of local ontologies.

Renaming of properties, comparing names of ontology properties with the corresponding terms in the synonym table, aims at ensuring consistency of domain terminologies and reusing the semantic data model in the field. The synonym table, which is constructed by domain experts and DBAs referring to exploration-development database handbooks, can solve problems of semantic heterogeneity. The names of terms with synonymous semantic relations in the handbooks are stored in a same collection in the synonym table. The collection name is unified into the corresponding name of the attribute in the entity, which is defined in 3.2.1.

If the name of ontology property is in the synonym table, rename the ontology property to the corresponding collection name in the synonym table. If it is not in the synonym table, user is required to complete the property renaming task through the GUI, and then add the property into the synonym table. If one property name of local ontology corresponds to multiple collection names in the synonym table, which is semantic heterogeneity of the same vocabulary expressing different meanings in different data sources, the GUI is also needed.

We propose a merging algorithm in the stage of classes merging. Comparing local ontology properties with the entity attributes constructed in the step of data filtering, the scope of ontology datatype properties of a class must be consistent with the corresponding attributes range of the entity, and the class name must be same with the corresponding entity name. If properties of two or more ontology classes correspond to one entity, merge the two or more classes into one class named the corresponding entity name.

The classes merging algorithm is detailed as follows.

Step1: Create an ontology class C_i , whose name is the name of entity $E(i)$.

Step2: $\forall DP_T \in C_T$, if $DP_T \in E(i) \wedge DP_T \notin C_i$, add DP_T into class C_i , and delete DP_T from class C_T . If $DP_T \in E(i) \wedge DP_T \in C_i$, delete DP_T from class C_T , and do not add DP_T into class C_i .

Step3: If $\forall DP_T \notin C_T$, delete class C_T , the C_T' constraint relationships convert into C_i' .

Step4: Traverse other classes C_T of local ontology, loop through Step 2 and 3.

Step5: Select other entities $E(i)$, and loop through Step 1-4 until all the entities have been traversed.

Figure 5 shows the normalized local ontology of production database after properties renaming and classes merging. Take class BlockInfo in Figure 5 as an example to illustrate the classes merging steps. Create a new class named BlockInfo firstly. In Figure 4, the names of datatype properties of class block_reservoir are in the entity BlockInfo, which is defined in the step of data filtering, so add the datatype properties into the new class BlockInfo, and delete the datatype properties from class block_reservoir. If all the datatype properties in class block_reservoir are deleted, delete class block_reservoir, and the constraint relationships of class block_reservoir are turned into class BlockInfo'. Similarly, traverse other classes. Here, we also add the datatype properties of block_physical into the new class BlockInfo.

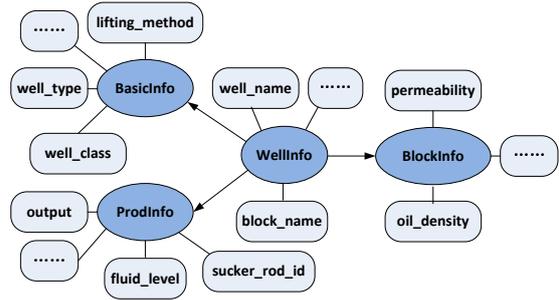


Figure 5: Normalized local ontology of production database (partial classes).

Next is combining local ontologies generated from different specialized databases into a global ontology. Starting to traverse the root classes of two local ontologies, if the two classes have the same datatype property, bridge the two classes by a foreign key constraint relationship. The class with functional property is converted into the subclass of the other class without functional property. Two local ontologies can be linked in this way. And then other local ontologies can be combined.

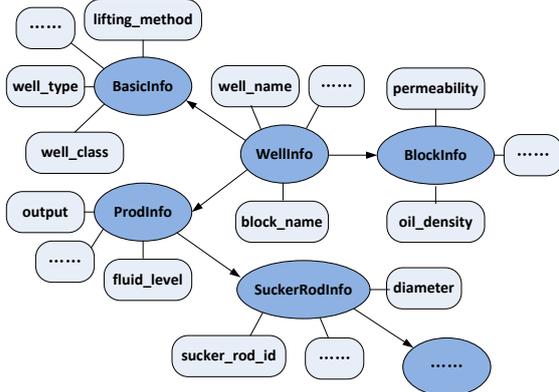


Figure 6: Global Ontology (partial classes).

Figure 6 shows a global ontology, which is a result of the combination of production database ontology and equipment database ontology. Sucker_rod_id is not only the primary key of table sucker_rod in equipment database, but also a property of table prod_info in production database, so bridge the two classes via sucker_rod_id by a foreign key constraint relationship.

Local ontologies can be converted into a global ontology after properties renaming, classes merging, and local ontologies combining.

3.2.4 Adding Semantic Constraint Rules

Semantic constraint rules are added to strengthen the

hierarchical relationships between concepts. Reasoning engine can use the constraint rules to reclassify and reorganize concepts of the global ontology, achieve a certain reasoning function, and obtain the implicit knowledge.

3.3 Semantic Query

According to the global semantic view, users can submit SPARQL statements to query the global ontology. SPARQL statements are converted into SQL to access the underlying data sources. Finally, the query results are presented to users in a uniform format after cleaning.

The semantic query implementation steps are as follows.

Step1: Get the query request, and generate the global query statement Q_G , which is described by SPARQL.

Step2: Reasoning engine converts names of classes/properties of Q_G in global ontology into the names in relative local ontologies based on the information of synonym table.

Step3: Divide the global query Q_G into sub queries $\{Q_{L1}, Q_{L2}, \dots, Q_{Ln}\}$ for local ontologies.

Step4: Rewrite sub queries $\{Q_{L1}, Q_{L2}, \dots, Q_{Ln}\}$ as local sub queries $\{Q_{D1}, Q_{D2}, \dots, Q_{Dn}\}$ for each data source. Local sub queries are described by SQL.

Step5: Execute local sub queries and return the results $\{R_{D1}, R_{D2}, \dots, R_{Dn}\}$ in unified formats.

Step6: Combine the results $\{R_{D1}, R_{D2}, \dots, R_{Dn}\}$, and return the final query response after data cleaning and converting.

4 APPLICATION OF SDSOGE

Due to the demand of oil and gas engineering domain, we develop the SDSOge system, which is implemented based on JAVA technology. SDSOge parses the global ontology and related local ontologies using Jena and makes the reasoning function into effect. Meanwhile, SDSOge implements the extraction of schemas of data sources and the data searching process using JDBC data access interfaces. SDSOge makes the use of data more profound and efficient.

Oil and gas engineering optimization design and assisted management system (OGEA) is a typical example of industrial application of SDSOge. OGEA is widely used in oil and gas engineering field. It could implement the production design and decision-making process with the support of

specialized databases, thus increase the production and recovery ratio.

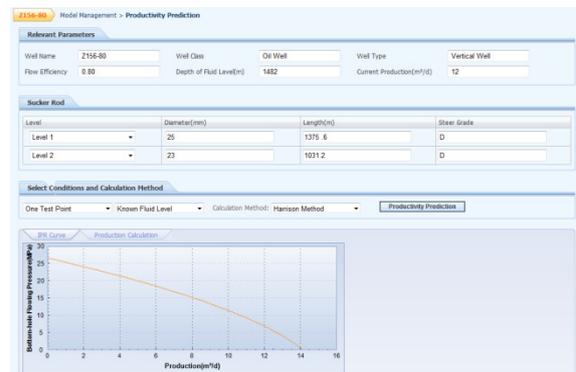


Figure 7: Interface of productivity prediction module.

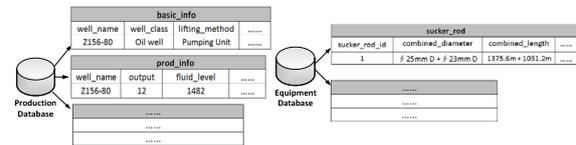


Figure 8: Corresponding data sources of productivity prediction module.

Figure 7 shows the interface of productivity prediction module of OGEA. The corresponding data sources of the module are shown in Figure 8. In Figure 7, the relevant parameters, such as depth of fluid level and current production, are collected from production database, while sucker rod data are collected from equipment database; which implements the integration of distributed data. The structure of sucker rod in Figure 7 is stored differently in databases from that in Figure 8. SDSOge shields the structural heterogeneity and presents sucker rod data to the upper level in the same format. The lower part of Figure 7 is the result of productivity prediction using the data in the upper portion. The application shown in Figure 7 is for multiple fields, but names of the same type of needed information are not identical in the databases of different oil fields. SDSOge can shield this semantic heterogeneity and map into the corresponding individuals by reasoning engine.

The OGEA system equipped with SDSOge has been put into production in oil fields of Daqing, Jilin, Huabei, and Dagang. Currently, SDSOge, which has measured effect evaluation for 28985 wells, could provide an entire and real-time data service of production monitoring and perform well in real applications.

After application of OGEA system with SDSOge in five oil production plants in Huabei Oil Field, the

average efficiency has increased by 3.6%, while the average pump inspection period has increased by 83 days, and total oil production has increased by 9054 tons. The cost of manpower and material resources has been saved, and the efficiency of management has been improved. Moreover, the average system efficiency has improved 3.75% and the average pump inspection period has increased by 75 days after the SDSOge applied in six oil production plants of Dagang Oil Field, which makes a lot of sense in extending pump inspection period, saving energy and raising production.

Based on the distributed and heterogeneous databases of oil fields, SDSOge shields the heterogeneity of underlying databases, builds the global semantic data model, provides the semantic searching function based on domain terminologies, and makes the searching results available for upper applications. SDSOge enables the value of data improved.

5 CONCLUSIONS AND FUTURE WORK

The current researches and applications mainly focus on solving semantic heterogeneity between data sources using ontology, data integration based on semantic methods, and data services for upper applications.

The semantic-based data service mentioned in this paper connects distributed, heterogeneous and complicated data seamlessly, which makes upper applications moving smoothly on SDSOge platform. SDSOge, which makes data shared and reused, builds a semantic-abundant global ontology in the domain of oil and gas engineering, implements data query transformations based on semantic methods, and provides a data service for upper applications. SDSOge could shield the heterogeneity of underlying data sources and allow users to access the standard data everywhere directly, thus provide effective data supports for production. SDSOge combines industrial production and scientific research tightly and is a great example that science promotes the progress of industry.

In the future, we would add more reasoning mechanisms to provide better semantic-based data services, and introduce SDSOge into more oil fields.

ACKNOWLEDGEMENTS

This work is supported by the R&D Infrastructure and Facility Development Program under Grant No. 2005DKA32800, the Key Science-Technology Plan of the National 'Twelfth Five-Year-Plan' of China under Grant No. 2011BAK08B04, the 2012 Ladder Plan Project of Beijing Key Laboratory of Knowledge Engineering for Materials Science under Grant No. Z121101002812005, the National Key Basic Research and Development Program (973 Program) under Grant No. 2013CB329606, and the Fundamental Research Funds for the Central Universities under Grant No. FRF-MP-12-007A.

REFERENCES

- Bellatreche, L., Dung, N. X., Pierra, G., Dehainsala, H., 2006. Contribution of ontology-based data modelling to automatic integration of electronic catalogues within engineering databases. In *Computers in Industry*, 57(8-9), 711-724.
- Carey, M. J., Onose, N., Petropoulos, M., 2012. Data Services. *Communications of the ACM*, 55(6), 86-97.
- Doan, A., Noy, N., Halevy, A., 2004. Introduction to the special issue on semantic integration. In *ACM SIGMOD Record*, 33(4), 11-13.
- Ghawi, R., Cullot, N., 2007. Database-to-Ontology Mapping Generation for Semantic Interoperability. In *Vldb '07*, Vienna, Austria.
- Halevy, A.Y., Ashish, N., Bitton, D., et al, 2005. Enterprise information integration: Successes, Challenges and Controversies. In *SIGMOD Conference*, 778-187.
- Hu, C., Tong, Z., et al, 2001. Research on Constructing of Object-Oriented Petroleum Common Data Model. *Journal of Software*, 12(3), 427-434.
- Kondylakis, H., Flouris, G., Plexousakis, D., 2009. Ontology and Schema Evolution in Data Integration: Review and Assessment. In *Meeraman, R., Dillon, T., Herrero, P. (eds.) OTM 2009. LNCS*, 5871, 932-947. Springer, Heidelberg (2009).
- Kondylakis, H., Plexousakis, D., 2011. Exlixis: Evolving Ontology-Based Data Integration System. In *SIGMOD'11*, 1283-1286.
- Ludäscher, B., Lin, K., Bowers S., et al, 2006. Managing Scientific Data: From Data Integration to Scientific Workflows. *Geological Society of America Special Paper on GeoInformatics*, 109-129.
- Mezini, M., Lieberherr, K., 1998. Adaptive Plug-and-Play Components for Evolutionary Software Development. In: *Proceedings OOPSLA'98*, ACM, Vancouver, British Columbia, Canada, 97-116.
- Ye, Y., Yang, D., Jiang, Z., Tong, L., 2008. Ontology-based semantic models for supply chain management. In *The International Journal of Advanced Manufacturing Technology*, 37(11-12), 1250-1260.

Cloud Space

Web-based Smart Space with Management UI

Anna-Liisa Mattila, Kari Systä, Jari-Pekka Voutilainen and Tommi Mikkonen

Department of Pervasive Computing, Tampere University of Technology, Korkeakoulunkatu 1, Tampere, Finland
{anna-liisa.mattila, kari.systa, tommi.mikkonen}@tut.fi, jari.voutilainen@iki.fi

Keywords: Internet-of-Things, Mobile Agents, HTML5, Web Applications, Experimentation.

Abstract: The emergence of HTML5 allows more complex applications to be run in browsers. However, these applications need not run inside the browser only. In our previous work we have shown that it is feasible to implement mobile agents with Web technologies, such as HTML5 and JavaScript. These mobile agents can be used to control systems like home automation. In this paper we show how this execution environment can be described as a Cloud Space that provides the users with a new type of multi-device experience to the content and the environment the users need to access and control. Furthermore, we present a new way to control and monitor the Cloud Space through a web application with a 3D UI based on direct manipulation.

1 INTRODUCTION

Modern web applications are systems that use dynamic HTML – HTML, CSS, DOM and JavaScript – for user interfaces and the HTTP protocol as the communication protocol. These systems form the backbone of the current ICT infrastructure. End users can easily access information and services through the Web using the browser without additional installation of specific client software. This commonality and uniformity has simplified end-users’ life, and given the browser a central role in all information access and nowadays increasingly often also in entertainment, office applications, and services such as banking.

For users, most of the relevant computing resources and content are located in the Internet. This trend is related to two recent developments: 1) users have multiple devices that they use to access data and services, and 2) users store their content in several cloud-based services. These two aspects are not yet fully supported since information that is typically local – like bookmarks – is often on a wrong device, and data that is stored in cloud-based services cannot be accessed as seamlessly as data stored in user’s own devices. Although some services provide a REST API – an architectural style for systems where resources play a major role (Fielding, 2000) – to access the content in a generic fashion, far too often data is only available through the specific service or proprietary application, and it is not possible to create new applications that would benefit from that data.

The increasing amount of computing power in our everyday environment is also an emerging trend. For example, home entertainment systems are increasingly often online, with considerable computing and storage capacity. We are surrounded by several “smart spaces”, or systems where our environment is controlled and monitored with computers and software running in them. Cheap computing devices, such as Raspberry Pi (Raspberry PI, 2013), have become widely available and are suitable for research and do-it-yourself (DIY) needs. This enables low-risk experiments and provides evidence that low-cost, Internet connected and powerful computing nodes are coming closer to us. We believe that the most natural way to access these pervasive computing elements are through a browser instead of platform specific installable proprietary solutions which are commonly used in today’s smart spaces.

In this paper we describe vision and proof-of-concept implementation of a system called Cloud Space, where all computing resources form interoperable clouds that users can access with any device. Users do not need to care about boundaries between particular services or individual devices available in smart spaces. Still, however, users can maintain their ownership and they are in control of their own content. In other words, data and computing will be completely cloudified for the user without risking users’ ownership of the content. In addition, we show how such an environment can be controlled and managed with a virtual 3D environment built using the

same technologies. In fact, the monitoring application could be stored in the same cloud infrastructure as the applications it is presently managing.

The rest of the paper is structured as follows. Section 2 describes the background. Section 3 introduces the concept of Cloud Spaces, and Section 4 describes a manager application for Cloud Spaces. Section 5 addresses related work. Finally Section 6 draws some final conclusions.

2 BACKGROUND

Our work has been motivated by a number of already reported research artifacts. These artifacts will be briefly introduced in the following subsections.

2.1 Interactive Web Applications

Emerging web technologies such as HTML5 (World Wide Web Consortium, 2012) and WebGL (Khronos Group, 2011) have rapidly altered the landscape of web application development. With such technologies, it is feasible to develop interactive applications with web technologies only, with no vendor-specific plug-ins that require separate installation. Even for games, the browser is increasingly often the desired platform due to its convenience – the users never need to install anything except the browser itself.

These interactive applications are changing our perception of browser-based applications. The code that runs in a browser is no longer a simple rendering procedure generated in the server-side. The applications are deployed and updated from the server like a web page, but after that the needs of the applications determine how dependent the execution is of the server. The applications can even be cached in the device so that they can be used in off-line situations. Examples of large web applications include systems like Google's Gmail¹, which consists of a considerable amount of code run inside the browser.

At the same time increasingly powerful libraries have simplified the development of JavaScript applications. These libraries help dealing with browser incompatibilities and JavaScript's not-so-good parts (Crockford, 2008). Furthermore, they support the development of web applications that provide the look and feel that is fundamentally similar to desktop applications. This was shown to be feasible by the Lively Kernel, which provided Smalltalk-like programming environment inside the browser (Ingalls et al., 2008), and Cloudberry (Taivalsaari and Systä,

2012) which showed that all user-visible software of a smart phone can be implemented with dynamically downloaded web-content.

2.2 Evolving Web Architecture

Latest web application trends have brought technologies originating from the browsers to servers, too. For example, Node.js (Node.js, 2013) has lately gained popularity. Node.js allows running JavaScript in the server side, and thus to some extent Node.js allows the execution of the same code in both server and client. This is a core enabler of our HTML5-agents described in the next subsection.

Technologies like Meteor (Meteor, 2013) and backend-as-a-service systems like Firebase (Firebase, 2013) are also changing the web architecture towards cloud-like architectures. In general, these systems provide features that are commonly needed online. Such features include user management, push notifications, and integration with social networking services, all of which are commonly needed in web applications.

2.3 HTML5 Agents

The Web is fundamentally based on mobile code, pages that may contain applications are downloaded from remote locations and evaluated inside the browser. Four paradigms of mobile code have been proposed (Carzaniga et al., 1997):

1. Client-Server where client uses code that is located in another node.
2. Remote Evaluation where client sends execution instructions, e.g. SQL queries, to another node.
3. Code on Demand where code is downloaded to the client for execution. HTML5 applications are widely used examples of code-on-demand.
4. Mobile Agent where code together with internal state of the application is moved to other node for execution.

The first three paradigms are regularly used in Web applications; the fourth paradigm is connected to cloud browsing paradigm (Taivalsaari et al., 2013) at least indirectly. In the following, we focus on the fourth paradigm, i.e. mobile agents.

In our previous work, we have designed an agent architecture (Systä et al., 2013), where agents can travel between hosts which can be either browsers or web servers. When running inside browsers these agents act like any Web application, but when located in a server, they are run in headless mode without a

¹<http://gmail.com/>

user interface. The agent can move between servers and browsers and thus toggle the mode between headless mode and UI mode. Furthermore, we support multi-device usage – the browser instance that pulls an executing agent can be different from the browser instance that had originally pushed the agent to the server. The internal state of the application is an important part of a mobile agent. HTML5 agent’s state needs to be serialized when the agent is moved from a location to another. Thus, the agent needs to be written so that serialization of the relevant parts of the state is possible. Our design provides support for such serialization. During agent’s life-cycle the agent may visit several browsers and several agent servers. A sample life cycle is presented in Figure 1.

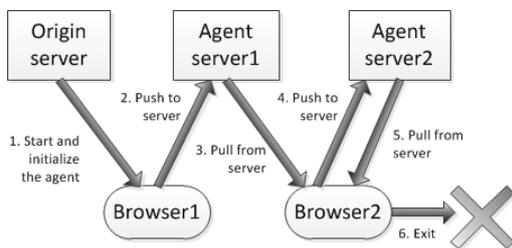


Figure 1: Life-cycle of a HTML5 agent (Systä et al., 2013).

In Figure 1, an agent is started by Browser1, when the agent is downloaded from its origin server (Step 1). In this phase the agent is initialized and the execution begins. Since the agent executes in a browser it has a user interface. In Step 2, the agent is pushed to an agent server. This means that the agent server gets the internal execution state of the agent and the application code (actually a URL to the code). The agent can continue the execution in the server. In Steps 3-5, the agent moves from one environment to another, but preserves its internal state and continues execution from where it left off. Finally, the execution is terminated in Step 6.

In our present implementation, the life cycle shown in Figure 1 has been modified so that agents can move between server and client but also between two servers. This liberates applications to travel freely between different computing nodes included in the system.

2.4 Agents for Web-of-Things

In (Järvenpää et al., 2013) we proposed using mobile agents in the context of Internet-of-Things (IoT). The approach was based on the fact that agent servers can be instantiated in small embedded computers and the mobile agents can move between different devices, and if necessary it is also possible to clone agents

to create more instances. One of the possible application areas we discussed in (Järvenpää et al., 2013) was home automation that goes beyond remote control when the agents can be run in embedded devices installed in the physical environment. In that case an intelligent agent can work on behalf of the user and implement even complex strategies to optimize energy consumption and user comfort.

There are approaches that are based on uploading and remote evaluation of code in a "thing". For example, MoteLab (Werner-Allen et al., 2005) is a test bed for sensor networks, where developers can upload executable Java to a device. Somewhat similar system is Kansei (Ertin et al., 2006) (later refactored to KanseiGenie) where developers can also create jobs to execute sensor applications. Our system can also be used in a similar way and from similar motivations. However, in our system the uploaded code is Web content and we can upload an executing agent with its internal state – i.e. our system is the paradigm of mobile agents.

Maybe the most similar approach to us is the mobile agent framework proposed in (Godfrey et al., 2013). It provides nodes in heterogeneous device networks with a way to communicate and co-operate. The system is based on Java-based AgentSpace (Silva et al., 1999) mobile agent platform. For us the use of Web technologies is essential since it enables leveraging the power of the web development ecosystem in application development (Systä et al., 2013). Furthermore, our agents have been designed to work a part of the Cloud Space explained in Section 3.

Since our mobile agents are based on Web technologies – HTML and JavaScript – they integrate well in Internet-based infrastructures. Moreover, normal browsers can be used to access and control the agents.

3 CLOUD SPACE

A key concept of our work is computing and content service called *Cloud Space*. Users have their own Cloud Spaces – in essence private clouds – in which all content is stored. Cloud Space includes the following three parts that relate and depend on each other:

1. A data solution that provides a uniform access to all content. Unlike typical cloud service, Cloud Space does not enforce service-specific silos or limit what kind of content is uploaded to the service. All the content is synchronized automatically between Cloud Space and devices and applications using the content.
2. A system that stores interaction (browsing) sessions so that they can be later continued in another

device.

3. The software infrastructure for mobile agents.

3.1 Concept

Cloud Spaces can be virtualized. For instance, a single computing resource can host several Cloud Spaces. However, like in cloud computing, the Cloud Spaces appear to both users and applications as a single service. The Cloud Space hosts arbitrary amount of *contexts*, which represent physical locations with devices like a kitchen, living room and mobile phone. Each context is a view to one or more Cloud Spaces with a user interface optimized for different screen sizes and for the needs of the context. Mobile phone context would need an easy access to lot of content in user's personal Cloud Space and Cloud Space of the current location but with simplistic UI since most likely, the screen size is small. Additionally to the context user interfaces, the admin interface needs its own UI. The admin UI handles creation of new users, new contexts and so on.

An example configuration of Cloud Space has been presented in Figure 2.

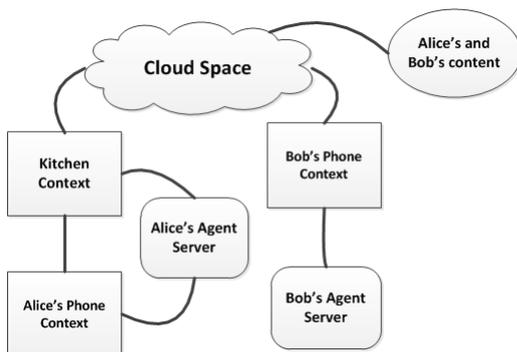


Figure 2: Concept of Cloud Space.

It should be noted that Cloud Space contexts can represent both physical and virtual entities. Furthermore, when user is in some physical space, for example in a room, she can access both her own personal Cloud Space and the Cloud Space of the room.

3.2 Implementation

Cloud Space is partly implemented as a proof-of-concept system. User management and context creation are implemented and they are available in the admin user interface. Currently contexts are bound to physical locations and agent servers can be assigned to a context. The design is implemented in Node.js,

mainly to leverage the existing agent server implementation. Hosting the user content will be exported to backend-as-a-service platform because synchronizing data across multiple databases is out of scope of the research.

4 MANAGEMENT UI IN 3D

As already pointed out, mobile agents can be used for managing and monitoring devices in the spirit of Internet-of-Things (Järvenpää et al., 2013). However, in our previous publications, (Systä et al., 2013), (Järvenpää et al., 2013) the management of the agents has been left as future work. In the following, we address the management of IoT environment where Cloud Space contexts and agents are located in the physical space.

4.1 Concept

The concept of the manager application is to provide a tool for a user to manage her Cloud Space contexts related to physical spaces, e.g. to a living room. Using the manager application the user can connect to the Cloud Space, select context and toggle between contexts. The user can monitor agent servers running in a context and move agents from a server to another server and to another context.

Because Cloud Space contexts are bound to physical spaces 3D user interface for the manager is a natural choice. A context can be visualized as a 3D model of the physical space to which the context is related. Resources related to the context and navigation widgets can be visualized as 3D objects.

Cloud Space and mobile agents use browsers as the front end. Similarly the monitor application is based on web technologies and runs on the browser.

4.2 Implementation

We are currently working on a proof-of-concept implementation of a Cloud Space context manager using WebGL and WebWidget3D (Mattila and Mikkonen, 2013) system. WebWidget3D is a 3D widget library that provides tools for creating 3D widgets and designing associated interaction. For actual rendering, WebWidget3D uses Three.js (Three.js, 2014) 3D engine for WebGL.

While the use of WebGL enables the creation of seemingly arbitrarily complex graphics and interactions, we simply aim at demonstrating the feasibility of our approach to the control and management tasks.

Consequently, the implementation effort has been invested in the creation of a simple model that can be easily understood and interacted with.

In the present implementation the user can manage her Cloud Space contexts using a direct manipulation 3D UI. Instead of using 3D models to visualize Cloud Space contexts, panoramic image spheres inside a 3D world are used. The agent servers and agents as well as navigation widgets are visualized using 3D objects. All user interaction is done using direct manipulation of 3D content. The user can move in the 3D world according to simple fly paradigm.

The manager consists of three views:

1. Login View where user logs in to her Cloud Space. This view is simple HTML form in 2D.
2. Context Selection View shown in Figure 3.
3. Context Management View shown in Figure 4.



Figure 3: Context selection view.



Figure 4: Context Management View.

When the user connects and logs in to her Cloud Space the manager fetches her contexts and moves to context selection view. The user can select the context to manage by double clicking a context sphere.

When the user has selected the context to monitor the Context Management View of the selected context is shown. The view consists of the servers related to the context visualized as grids inside the context sphere. Boxes inside grids are representations of agents. The user can load an agent to her browser

by double clicking an agent box. Moving an agent to another server can be done by dragging the agent to another grid and dropping it there.

Inside the Context Management View the Selection View is also visualized. The user can toggle between Context Management Views using the Selection View. The Selection View inside a Context Management View is shown in right side of Figure 4. The user can also drag agents from a context to another. Moving agents are visualized in Figure 5.



Figure 5: Moving agent from a Context to another Context.

In the first fragment of Figure 5 (marked with number 1) the user has dragged the agent on top of the context which she wants to move the agent. When she releases the agent the Management View changes to the context she chose. This is visualized in the fragment 2 of Figure 5. Finally the user can drag the agent to the server in the context and the agent is moved there (fragment 3 in Figure 5).

5 RELATED WORK

As far as we know our HTML5 mobile agents is a unique application of HTML technology. However, mobile agents as such are an old invention of technologies and concepts (Carzaniga et al., 1997). Similarly, Cloud Space is a unique combination, although parts of it have been realized by commercial systems like iCloud by Apple, which implements some parts of the Cloud Space's idea but is limited to a vendor-specific "silo".

Visualization of the agent configuration is implemented, for example, in Motelab (Werner-Allen

et al., 2005) where nodes in a sensor network can host Java-based agents. In MoteLab a Maps Page shows the sensor network with connections, but the visualization is two-dimensional only and does not include management operations. Some management aspects have been included in the WebIoT architecture (Castellani et al., 2012), where a heterogeneous device set can be visualized and controlled through an extensible user interface. Similarly to our approach the UI of WebIoT is based on the Web and runs in a browser. However, WebIoT does not use 3D to show the UI and does not introduce direct manipulation.

6 CONCLUSIONS

In this paper we have introduced the concept of Cloud Spaces and a proof-of-concept 3D interface for managing Cloud Spaces in the IoT context.

The concept of Cloud Spaces provides personal clouds for user, but it is more than data storage. A Cloud Space can also host agent servers and a Cloud Space context can be connected to a physical space. This makes it possible to use Cloud Spaces together with HTML5 agents to build smart spaces for e.g. home automation.

The 3D UI for managing contexts inside a Cloud Space shows how a browser can host interactive 3D user interface for monitoring and managing smart spaces. The UI provides functionality, e.g. for moving agent from server to another server, with direct manipulation in the 3D context. Exploring the most feasible 3D visualizations for contexts, agent servers and agents in a smart space are at this point left as future work.

REFERENCES

- Carzaniga, A., Picco, G. P., and Vigna, G. (1997). Designing Distributed Applications With Mobile Code Paradigms. In *Proceedings of the 19th international conference on Software engineering*, pages 22–32. ACM.
- Castellani, A. P., Dissegna, M., Bui, N., and Zorzi, M. (2012). WebIoT: A Web Application Framework for the Internet of Things. In *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE*, pages 202–207. IEEE.
- Crockford, D. (2008). *JavaScript: The Good Parts*. O'Reilly.
- Erting, E., Arora, A., Ramnath, R., Nesterenko, M., Naik, V., Bapat, S., Kulathumani, V., Sridharan, M., Zhang, H., and Cao, H. (2006). Kansei: A Testbed for Sensing at Scale. In *Proceedings of the 4th Symposium on Information Processing in Sensor Networks (IPSN/SPOTS TRACK)*, pages 399–406. ACM Press.
- Fielding, R. T. (2000). *Architectural Styles and the Design of Network-Based Software Architectures*. PhD thesis, University of California.
- Firebase (2013). Web page of firebase technology. Technical report. <https://www.firebase.com/>. Last viewed 31.12.2013.
- Godfrey, W. W., Jha, S. S., and Nair, S. B. (2013). On a Mobile Agent Framework for an Internet of Things. In *Proceedings of the 2013 International Conference on Communication Systems and Network Technologies, CSNT '13*, pages 345–350, Washington, DC, USA. IEEE Computer Society.
- Ingalls, D., Palacz, K., Uhler, S., Taivalsaari, A., and Mikkonen, T. (2008). The Lively Kernel a Self-Supporting System on a Web Page. In *Self-Sustaining Systems*, pages 31–50. Springer.
- Järvenpää, L., Lintinen, M., Mattila, A.-L., Mikkonen, T., Systä, K., and Voutilainen, J.-P. (2013). Mobile Agents for the Internet of Things. In *System Theory, Control and Computing (ICSTCC), 2013 17th International Conference*, pages 763–767. IEEE.
- Khronos Group (2011). WebGL Specification. Technical report. <http://www.khronos.org/registry/webgl/specs/1.0/>.
- Mattila, A.-L. and Mikkonen, T. (2013). Designing a 3D Widget Library for WebGL Enabled Browsers. In *proceedings of the 28th Symposium On Applied Computing*, volume 1, pages 757–760. ACM.
- Meteor (2013). Web page for meteor technology. Technical report. <https://www.meteor.com/>, Last viewed 31.12.2013.
- Node.js (2013). Web page for document and download of node.js technology. Technical report. <http://nodejs.org/>. Last viewed 31.12.2013.
- Raspberry PI (2013). Web page of raspberry pi. Technical report. <http://www.raspberrypi.org/>, last viewed 31.12.2013.
- Silva, A., Silva, M. M. d., and Delgado, J. (1999). An Overview of AgentSpace: A Next-Generation Mobile Agent System. In *Proceedings of the Second International Workshop on Mobile Agents, MA '98*, pages 148–159, London, UK, UK. Springer-Verlag.
- Systä, K., Mikkonen, T., and Järvenpää, L. (2013). HTML5 Agents - Mobile Agents for the Web. In *WEBIST*, pages 37–44.
- Taivalsaari, A., Mikkonen, T., and Systä, K. (2013). Cloud Browser: Enhancing the Web Browser With Cloud Sessions and Downloadable User Interface. In *Grid and Pervasive Computing*, pages 224–233. Springer.
- Taivalsaari, A. and Systä, K. (2012). Cloudberry: An HTML5 Cloud Phone Platform for Mobile Devices. *Software, IEEE*, 29(4):40–45.
- Three.js (2014). Web page of three.js 3d engine. Technical report. <http://threejs.org/>, last viewed 7.1.2014.
- Werner-Allen, G., Swieskowski, P., and Welsh, M. (2005). MoteLab: A Wireless Sensor Network Testbed. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 68. IEEE Press.
- World Wide Web Consortium (2012). HTML5 Specification, candidate recommendation. Technical report. <http://www.w3.org/TR/html5/>.

Sequential Model of User Browsing on Websites

Three Activities Defined: Scanning, Interaction and Reading

Aneta Bartuskova and Ondrej Krejcar

*Dpt. of Information Technologies, University of Hradec Kralove, Rokitanskeho 62, Hradec Kralove, Czech Republic
aneta.bartuskova@uhk.cz, ondrej@krejcar.org*

Keywords: Websites, Interaction, Browsing, Usability, Aesthetics, Information Quality.

Abstract: This paper presents a model of user browsing behaviour on websites. Main user activities on websites are suggested, discussed and supported by previous research. Proposed activities are then associated with three main aspects of the website - usability, aesthetics and information quality. Their role in each phase of user browsing on the website is discussed. Basic browsing model is then constructed on the basis of previous research's conclusions, accompanied by new considerations. Model variations are taken into consideration and discussed in relevance to the mode of use.

1 INTRODUCTION

User browsing, interaction and generally behaviour on website are widely researched topics in human-computer interaction, which can be studied in various contexts and from many different angles. Many of research goals in this area eventually lead to user preference, which is very important in today's competitive environment. User preference, user experience and evaluation in the scope of websites are often associated with constructs like usability and aesthetics.

Main goal of this paper is to connect these constructs or aspects with phases of interaction between a user and a website. According to authors, every phase has its prominent aspect, which has the biggest influence on user. Proposition of these activities is supported by review of relevant literature. Browsing model of user activities on the website is then constructed, on the basis of previous research and new considerations about expected course of actions.

2 ASPECTS OF THE WEBSITE

The use of a webpage is determined by several factors: the information provided, usability of the website and the impression given to the user (Schenkman and Jönsson, 2000). Web design

attributes were defined as: content organization, visual organization, navigation system, colour and typography (McCracken and Wolfe, 2004). Websites can be evaluated by their usability, memorability, aesthetics, information quality and engagement, which result in overall preference (de Angeli, Sutcliffe and Hartmann, 2006).

Generally, three main aspects of websites emerge from previous research: usability, aesthetics and content (or information quality).

2.1 Aesthetics

Aesthetics of user interfaces is undoubtedly one of the most influential factors of their success with users. General concept of aesthetics comprises several similar constructs such as visual appeal, beauty or goodness.

Beauty is an important predictor of the overall impression and user judgment and therefore beauty of a webpage is an important factor determining how it will be experienced and judged (Schenkman and Jönsson, 2000). Another research showed an influence of aesthetics on credibility and trust, dependent mainly on first aesthetics impression of the website (Robins and Holmes, 2008). Other construct similar to aesthetics - perceived visual attractiveness of the website - was proven to influence usefulness and ease-of-use, i.e. usability (van der Heijden, 2003).

2.2 Usability

Usability can be taken as an objective construct (precise measurements of user performance) or subjective (perceived usability). This division is similar to another two concepts: pre-use usability, which is perceived usability of the interface before use, and user performance as a result of user's activities on the site (Lee and Koubek, 2010). As specified in ISO 9241-11, we can also divide usability measures into these three groups: the measures of effectiveness, efficiency and satisfaction (Hornbæk, 2006).

There is not a conformity among various studies, which aspects are included in usability. One study presents as usability criteria: ease of use, readability, productivity, content quality, completeness or relevance (Spool et al., 1999). Other extensive research includes consistency, navigability, supportability, learnability, simplicity, interactivity, telepresence, credibility, content relevance and readability (Lee, Kozar, 2012).

According to authors' opinion, content should create a separate category, along with its attributes such as content quality, content relevance or completeness. Usability aspect of websites should be limited to ease of use according to layout, navigation, affordances, readability and similar concepts.

2.3 Content

Finally, content or information quality is one of the key aspects in a website's success (Lynch and Horton, 2001). Characteristics of content can be defined as quality and quantity of provided information (de Angeli et al., 2006). Content can be also taken as a subjective measure in form of perceived quality of content (Bartuskova and Krejcar, 2013).

Content is often presented as part of usability aspect, nevertheless it creates a whole different category. Content's criteria relevant to textual form can be divided into quantity measures (e.g. completeness) and quality measures (e.g. relevance, accuracy or understandability).

It is however apparent, that these aspects - aesthetics, usability and content - can be taken separately only to a certain degree. They are all present together in the website, they have an influence on each other and they are all incorporated in overall user preference. Relation between usability and aesthetics in human-computer interactions generally is widely researched (Tuch,

Roth, Hornbæk, Opwis, Bargas-Avila, 2012). Previous studies have shown that subjective evaluations of usability and aesthetics are correlated (Hassenzahl, 2004).

3 ACTIVITIES ON WEBSITE

There is a large number of studies, which deal with browsing and interaction on websites. To the authors' best knowledge, there is limited research on sequential modelling of user activity in scope of one website and one session, in association with website's aspects. The approach presented in this paper aspires to bring a novel view on this matter.

Three essential user activities were identified in relation to browsing a website. These activities were labelled as: scanning, interaction and reading. Scanning comprises visual scan of a website, along with developing basic orientation on the website, scanning text and pictures and building first impression. Interaction includes searching for interaction possibilities and using them in actual interaction with the website. Reading activity involves more thorough scanning and actual reading of website content, as well as its understanding and evaluation. Suggested activities are based on review of related literature, which is further discussed in separate sections.

3.1 Scanning

Scanning had been used in literature as e.g. organizational scanning or browsing. Four scanning modes had been defined: undirected viewing, conditioned viewing, informal search and formal search (Choo et al., 2000). These strategies are divided according to mode in which user access internet. Scanning can be of different nature according to user's mode of browsing. Users can either look up a certain web page for a particular piece of information or just surf the internet without any particular goal of their surfing (Schenkman and Jönsson, 2000). Scanning in this context indicates user behaviour across more websites. Scanning in the context of this article refers to brief survey of the website (one website) visually, also including basic text scanning and searching for affordances.

In a scenario of one website, user can either search for particular information within the website, or he can just browse through the website according to what catches his attention. Moving to another webpage is very easy if the current webpage does not appeal to the user, which is why the first

impression of websites is so important (Schenkman and Jönsson, 2000). It was proven that people form an opinion about website based on its visual appeal in a time interval as short as 50ms (Lindgaard et al., 2006).

Scanning is, according to authors' opinion, the first activity performed by a user while entering a website and includes several continuously proceeding actions:

- gathering impression about visual appeal (usually mostly unaware)
- scanning graphics and pictures - according to their nature, pictures either contribute to visual appeal or help the user with scanning text or searching for affordances
- scanning text - searching for desired text fragments in headlines and paragraphs
- searching for affordances (action possibilities) - hyperlinks, menu items, and other interactions

Studies of how users read on websites found that they do not actually read, instead they scan the text, or they first scan the text before actually reading it (Morkes and Nielsen, 1997). Scanning text means not reading word by word, but e.g. only the first sentence of each paragraph to find the desired information. If the user finds scanned section of text satisfactory, reading activity takes place. If the user does not find desired information, he tries to interact with the website, usually in order to get to another set of information. Usually it requires at least several mouse clicks until the user finds what he is looking for. Interaction is therefore the next activity in proposed browsing model.

3.2 Interaction

Interaction in the context of this article means finding and using an affordance (action possibility) on the website, which is conditioned by quality of the information architecture and navigation of the website.

Affordances are not just about functional meanings and motor capabilities; they are also about emotional and cognitive processes that emerge through interaction (Overbeeke and Wensveen, 2003). Interaction aesthetics are one among other factors that allow users to enhance the detection of action possibilities and consequently, the detection of affordances (Xenakis and Arnellos, 2013). That is why searching for affordances is included also in scanning activity and it is therefore connected with aesthetics, even though interactions are mostly associated with usability.

Affordances include control areas of the website such as menu, hyperlinks in sections of text, additional functionality in the form of buttons etc. Interaction activity implies finding desired functionality and appropriately using associated affordance.

This interaction activity comprises several actions:

- searching for functionality (this originate from scanning activity)
- identifying desired affordance
- using the affordance correctly (e.g. hovering or clicking)

Successful search for functionality is dependent on purposeful navigation and logical information architecture. It also depends on design and therefore also aesthetics, or more specifically interaction aesthetics. Also successful identification of the affordance and using it correctly depends on appropriate design. Correct usage implies recognition of action - usually it is a mouse clicking, but it can be also e.g. hovering, dragging or scrolling.

3.3 Reading

Reading activity is proposed to follow after scanning activity, as users usually scan the text before actually reading it. Reading activity can be preceded by series of scanning and interaction activities, until the user reaches desired or just interesting section of text. Reading is expected to include two sequentially or interchangeably performed actions:

- more thorough scanning of headlines and paragraphs
- actual reading and understanding of the text
- retrieving desired information
- evaluation of read text and retrieved information

Successful reading depends on many factors associated with information quality and quantity but also usability, especially legibility. Content should be relevant, understandable and its arrangement should follow some basic design principles such as chunking or proximity.

All presented activities are performed by user sequentially, some of them are overlapping in specific actions.

4 BROWSING MODEL

User browsing cannot be entirely generalized, as every user has different background, abilities,

personality etc. which results in individual browsing style. Nevertheless general order of actions can be expected based on conclusions from previous sections and related literature with performed experiments on user testing.

The authors suppose, that in every sequential phase or activity, the different aspect of the website is primarily influencing user actions and also success of his actions.

It was demonstrated that visual appeal or aesthetics is likely to be detected first and it can influence subsequent experience with the webpage (Lindgaard, Fernandes, Dudek and Brown, 2006). Therefore scanning as the first activity is supposed to be mostly connected with aesthetics aspect. Aesthetics is then the most pronounced during the scanning activity.

Interaction activity includes searching for interaction possibilities and their usage. This activity is influenced especially by usability of user interface such as information architecture, navigation etc. Therefore, usability is expected to be the most pronounced aspect of interaction activity.

Reading activity is supposed to be experienced at the latest, as the user rarely finds what he is looking for on the first page. Information quality (or quality of content) is proposed as the most influential during reading activity.

Table 1: Suggested user activities on the website associated with most pronounced aspects.

| User activity | The most pronounced aspect during the activity |
|---------------|--|
| Scanning | Aesthetics |
| Interaction | Usability |
| Reading | Content |

4.1 Simulation of User Browsing

Development of activities in the presented simulation is an example of real situation, when the user starts actual reading or information retrieval after two clicks and then again and again after additional click. The magnitude of individual aspects signifies their participation on current activity, which is perceived by the user. Actual values are estimated according to previous research conclusions and also authors' own presumptions, which are listed in the next section.

4.2 Entering Conditions

Previous simulation depicts expected influence of aesthetics, usability and content on user in different

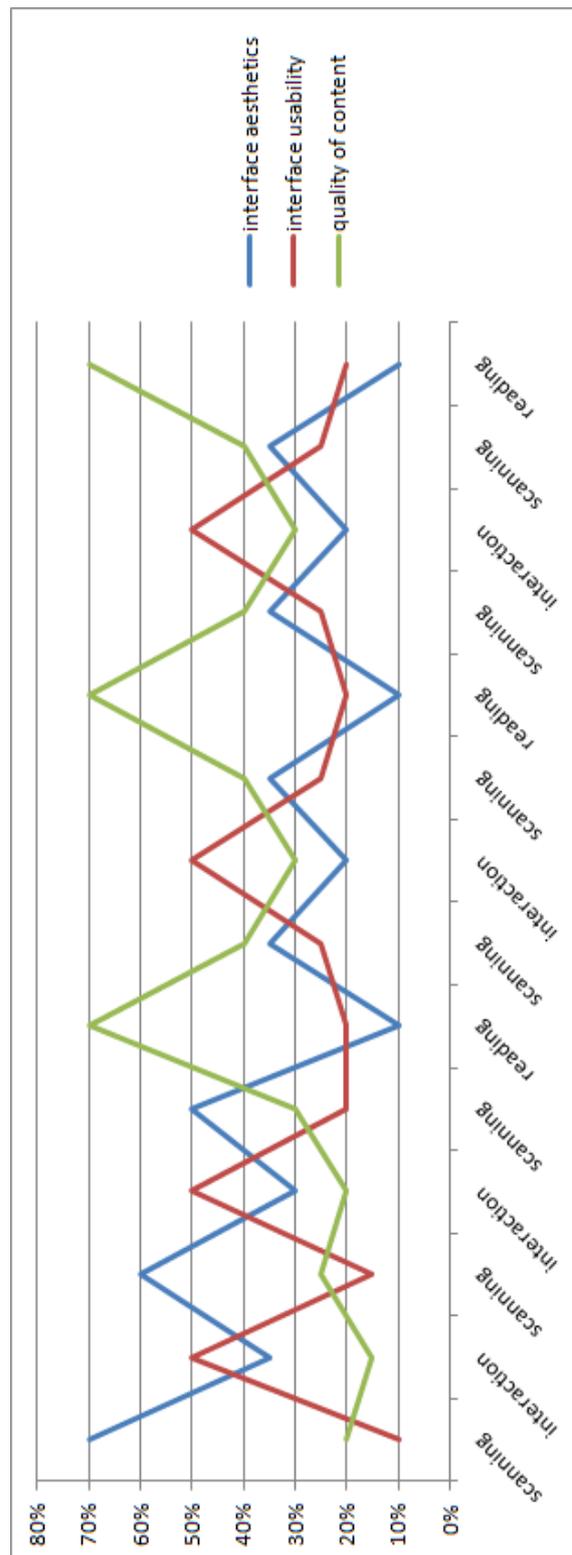


Figure 1: Simulation of user browsing on the website.

phases of working with the website. The conditions on which was constructed simulation of the browsing model are discussed in this section. These presumptions are:

- aesthetics is the most influential aspect in the scanning activity (see first aesthetics impression), but it is gradually losing its magnitude (only to a certain degree - feeling on visual appearance usually persists) with user's increasing interest in content, which can be expected with further browsing on the website
- aesthetics, especially in a form of interaction aesthetics, is also significant during interaction activity, but its magnitude is again gradually decreasing (only to a certain degree)
- aesthetics during actual reading or information retrieval is quite insignificant on stable level
- usability is the most prominent aspect during the interaction activity and its share of influence is expected to be stable during all interactions
- influence of usability in scanning activity is low at first (aesthetics dominates), but is gradually increasing, as the usability issues such as visual organization and navigation are becoming more apparent during scanning
- participation of usability while actual reading is low but higher than of aesthetics, as organization and legibility are parts of usability aspect
- quality of content is of course most significant during reading activity
- quality of content in scanning and interaction activities is low at first but gradually increasing, as orientation on the website is already clearer for the user and visual impression is established, scanning is expected to become more content-oriented with more time spent on the website

Presented model and its development suggest, that influence of individual aspects and their participation on overall judgement is varying according to the time spent browsing on the website and distribution of performed activities in that time. It is expected, that with more time spent browsing the website, the overall judgement will be more influenced by information quality.

The aesthetics and usability aspects however are crucial for actual getting to the content. This usually corresponds with the real situation. Another study which was researching importance of quality dimensions to overall judgement also discovered that the most important component was content, then

usability and finally aesthetics (Hartmann et al., 2007).

4.3 Mode of Use Variations

The overall judgement as well as perception of usability and aesthetics are highly dependent on context (de Angeli, Sutcliffe and Hartmann, 2006). They are also influenced by the mode in which the user approaches the interface (van Schaik and Ling, 2009). Information retrieval is different than surfing. When looking for information, users are more focused and content is the driving force. When users surf, they are just browsing and clicking at what looks most interesting (Spool et al., 1998).

Mode of use would certainly influence the browsing model. It is expected that aesthetics would be more influential during surfing. In the goal mode could be more significant usability and information quality even during scanning.

5 CONCLUSIONS

This paper presented the model of user browsing behaviour on the website. Main user activities during browsing on the website were proposed, discussed and supported by related literature. Suggested activities were associated with three main aspects of the website, which were identified as aesthetics, usability and information quality. Associated in the sense that they are primarily influencing user actions and also success of his actions during the relevant activity.

Browsing model was designed on the basis of previous research's conclusions and new considerations. Simulation of user browsing on the website was presented. Variations of the model according to mode of use were discussed.

More factors can alter course of the browsing simulation. Proposed browsing model dealt only with the first visit on the website in one browsing session. The influence of individual aspects would be different in case of repeated visits. The model also did not take into consideration various types of websites. This should be covered in future research as well as supported with results from user testing.

ACKNOWLEDGEMENTS

This work and the contribution were supported by: (1) project No. CZ.1.07/2.2.00/28.0327 Innovation

and support of doctoral study program (INDOP), financed from EU and Czech Republic funds; (2) project “Smart Solutions in Ubiquitous Computing Network Environments”, from the Grant Agency of Excellence, University of Hradec Kralove, Faculty of Informatics and Management, Czech Republic; (3) project “SP/2014/ - Smart Solutions for Ubiquitous Computing Environments” from FIM, University of Hradec Kralove.

REFERENCES

- Bartuskova, A., Krejcar, O., 2013. Evaluation framework for user preference research implemented as web application. In *Computational Collective Intelligence, Technologies and Applications, Lecture Notes in Computer Science*, 537-548.
- de Angeli, A., Sutcliffe, A., Hartmann, J., 2006. Interaction, usability and aesthetics: what influences users' preferences? In *Proceedings of the 6th conference on Designing Interactive systems*, University Park, PA, USA.
- Dondio, P., Longo, L., Barrett, S., 2008. A translation mechanism for recommendations, In *International Federation for Information Processing*, Vol. 263, pp. 87-102.
- Hartmann, J., Sutcliffe, A., de Angeli, A., 2007. Investigating attractiveness in web user interfaces. In *Proceedings of the SIGCHI conference on human factors in computing systems*, 396.
- Hassenzahl, M., 2004. The interplay of beauty, goodness, and usability in interactive products. In *Human-Computer Interaction*, 19(4), 319-349.
- Hornbæk, K., 2006. Current practice in measuring usability: Challenges to usability studies and research. In *International Journal of Human-Computer Studies*, 64 (2).
- Choo, C. W., Detlor, B., & Turnbull, D., 2000. Information seeking on the Web: An integrated model of browsing and searching. Available from: http://firstmonday.org/issues/issue5_2/choo/index.html.
- Krejcar, O., 2007. Benefits of building information system with wireless connected mobile device - PDPT Framework. In *1st IEEE International Conference on Portable Information Devices, IEEE Portable 2007*, March 25-29, 2007, Orlando, USA. pp. 251-254. DOI 10.1109/PORTABLE.2007.57.
- Krejcar, O., Jirka, J., Janckulik, D., 2011. Use of Mobile Phone as Intelligent Sensor for Sound Input Analysis and Sleep State Detection. In *Sensors*. vol. 11, Iss. 6, pp. 6037-6055. DOI 10.3390/s110606037.
- Lee, S., Koubek, R. J., 2010. The effects of usability and web design attributes on user preference for e-commerce web sites. In *Computers in Industry*, 61 (4).
- Lee, Y., Kozar, K. A., 2012. Understanding of website usability: Specifying and measuring constructs and their relationships. In *Decision Support Systems*, v.52 n.2.
- Lindgaard, G., Fernandes, G., Dudek, C., Brown, J., 2006. Attention web designers: you have 50 milliseconds to make a good impression! In *Behaviour and Information Technology* 25, 115-126.
- Longo, L., Barrett, S., Dondio, P., 2009. TOWARD SOCIAL SEARCH From Explicit to Implicit Collaboration to Predict Users' Interests, In *proceedings of 5th International Conference on Web Information Systems and Technologies*, pp. 693-696.
- Lynch, P. J., Horton, S. (2001). *Web Style Guidelines (2nd ed.)*. Yale University Press.
- McCracken, D. D., Wolfe, R. J., 2004. *User-centered Website Development: A Human-Computer Interaction Approach*. Pearson Prentice Hall Inc., Upper Saddle River.
- Morkes, J., & Nielsen, J., 1997. Concise, SCANNABLE, and Objective: How to Write for the Web. Available from: <http://www.useit.com/alertbox/9710a.html> and <http://www.useit.com/papers/webwriting/writing.html>.
- Overbeeke, K. C. J., & Wensveen, S., 2003. From perception to experience, from affordances to irresistibles. In *Proceedings of the 2003 international conference on designing pleasurable products and interfaces* (pp. 92 e 97). Pittsburgh, PA, USA: ACM.
- Penhaker, M., Jeziorska R., Novak, V., 2013. Computer Based Psychometric Testing and Well Being Software for Sleep Deprivations Analysis. In *Studies in Computational Intelligence*, vol. 457. pp. 207-216.
- Robins, D., Holmes, J., 2008. Aesthetics and credibility in web site design. In *Information Processing & Management*, 44 (1), 386-399.
- Schenkman, B., Jönsson, F., 2000. Aesthetics and preferences of web pages. In *Behaviour and Information Technology*, 19(5), 367-377.
- Spool, J., Scanlon, T., Schroeder, W., Snyder, C., DeAngelo, T., 1998. *Web Site Usability: A Designer's Guide*. Morgan Kaufmann Publishers, San Francisco.
- Tuch, A. N., Roth, S. P., Hornbæk, K., Opwis, K., Bargas-Avila, J. A., 2012. Is beautiful really usable? Toward understanding the relation between usability, aesthetics, and affect in HCI. In *Computers in Human Behavior*, 28(5), 1596-1607.
- van der Heijden, H., 2003. Factors influencing the usage of websites: the case of a generic portal in the Netherlands. In *Information & Management* 40 (6).
- van Schaik, P., Ling, J., 2009. The role of context in perceptions of the aesthetics of web pages over time. In *International Journal of Human-Computer Studies*, 67(1).
- Xenakis, I., Arnellos, A., 2013. The relation between interaction aesthetics and affordances. In *Design Studies*, 34 (1), 57-73.

Automated Usability Testing for Mobile Applications

Wolfgang Kluth, Karl-Heinz Krempels and Christian Samsel
Information Systems & Databases, RWTH Aachen University, Aachen, Germany
{kluth, krempels, samsel}@dbis.rwth-aachen.de

Keywords: Usability Testing, Usability Evaluation, HCI, Automated Testing, Mobile.

Abstract: In this paper we discuss the design and implementation of an automated usability evaluation method for iOS applications. In contrast to common usability testing methods, it is not explicitly necessary to involve an expert or subjects. These circumstances reduce costs, time and personnel expenditures. Professionals are replaced by the automation tool while test participants are exchanged with consumers of the launched application. Interactions of users are captured via a fully automated capturing framework which creates a record of user interactions for each session and sends them to a central server. A usability problem is defined as a sequence of interactions and pattern recognition specified by interaction design patterns is applied to find these problems. Nevertheless, it falls back to the user input for accurate results. Similar to the problem, the solution of the problem is based on the HCI design pattern. An evaluation shows the functionality of our approach compared to a traditional usability evaluation method.

1 INTRODUCTION

In a time when more and more consumers use technical devices to manage their everyday life, usability in software is important. A user friendly handling of smart phones, personal computers and smart televisions depends on the interface between human and computer. The large number of mobile applications in the consumer market leads to increased efforts to improve the usability for mobile devices. Additionally, better and faster hardware leaves mobile devices more capabilities for realizing complex software, but the device and display is still of small size. Thus, it is a challenge to develop appropriate software with good user experience.

In human-computer interaction (HCI) one goal is measuring and improving the usability of soft- and hardware. Different usability testing methods have been developed to estimate the quality of user interfaces and to derive solutions for usability improvements. While evaluations with users (e.g., cognitive walkthrough and heuristic evaluation) and without users (e.g., user observation and think aloud) are widely used, automated usability testing (AUT) is still an untouched area, especially in the method of mobile devices. With and without users, usability testing requires a lot of development time, money and HCI experts. These are reasons and excuses to avoid the integration in software development processes. One so-

lution could be the automation of usability tests with the objective of reducing the efforts of software developers to make usability evaluation more attractive.

The focus of this work is on mobile applications, which, in comparison to personal computers, have additional usability problems due to their mobile context (i.e., in which situation the device is used) (Schmidt, 2000), size and computing power. Nevertheless, it is also a challenge to find appropriate usability testing methods to evaluate mobile applications (Zhang and Adipat, 2005). The development of the mobile device and app market shows the importance of an automated usability evaluation tool.

Our main objective for this paper is the development of a fully automated tool for testing usability problems of mobile applications in the post-launch phase implemented for Apple's¹ iOS platform. It should replace the current evaluation technique *think aloud* which is typical for smart phones.

This paper is structured in six chapters. Section 2 gives an overview of current approaches in AUT. Section 3 and Section 4 describe the theoretical idea and implementation. At the end, in Section 5 the implementation is tested with a bike sharing application prototype and Section 6 reviews this work with an outlook to future work.

¹<http://www.apple.com>

2 RELATED WORK

In (Ivory and Hearst, 2001) AUT is separated in five method classes: *testing*, *inspection*, *questionnaire*, *analytic models*, and *simulation*. The approach described in this paper concentrates on the class of testing with real users which are involved to evaluate the mobile application. However, we limit the amount of related work in this section to mobile platforms.

According to (Ivory and Hearst, 2001), AUT has four different steps of automation: *nonautomatic*, *automatic capture*, *automatic analysis*, and *automatic critic*. Each automation step is consecutive to its predecessor. Furthermore, the effort of AUT is estimated formally (explicit tasks for participants) and informally (participants use target system without any further tasks).

2.1 Capturing

In (Lettner and Holzmann, 2012a) and (Lettner and Holzmann, 2012b), Lettner and Holzmann present their *Evaluation Framework* for capturing user interactions for the Android² environment. The approach works with aspect-oriented programming (AOP) which adds the capture functionality automated within the Android application. With AOP the compiler includes the important logger methods directly into the application's lifecycle.

Another capturing approach is presented in (Weiss and Zduniak, 2007) where a capture and replay framework for Java2MicroEdition (J2ME) environment has been developed. A proxy was used in combination with *code injection* (modify event methods) to intercept graphical user interface (GUI) events. The tool creates a log file, which can be read, modified by the developer, and replayed on a simulator.

2.2 Analysis

Many approaches for websites and desktop computers exist which automatically capture the interactions of the users to analyze them (Ivory and Hearst, 2001). Nevertheless, their functionality is reduced to the visualization of interaction processes and statistical analyses of e.g., resting time in views, number of clicks, and hyperlink selections.

One of the rare AUT tools for analysis of mobile applications is *EvaHelper* (Balagtas-Fernandez and Hussmann, 2009). It is implemented in a four phase model: *preparation*, *collect*, *extraction*, and *analysis*. *Preparation* and *collect* are phases of the automated capturing process and in contrast to approaches

²<http://www.android.com>

in Section 2.1 it needs a manual implementation of the logger methods. Nonetheless, in *extraction* the log is converted into GraphML³, a machine readable format. This format makes it possible for the developer to apply explicit queries on this graph for analysis.

2.3 Critic

In (Albraheem and Alnuem, 2012) a survey of AUT approaches shows that only one approach exists which implements *automatic critic* for mobile applications. With *HUI Analyzer* (Au et al., 2008)(Baker and Au, 2008) Au and Baker developed a framework and an implementation for capturing and analyzing user interactions for applications of the Microsoft .NET Compact Framework 2.0 environment. *Automatic critic* is implemented in this project in an automatic review of static GUI elements on the basis of guidelines. However, with *HUI Analyzer* it is possible to compare actual interaction data (e.g., clicks, text input, and list selections) with expected interaction data (i.e., series of interactions predefined by the evaluator). With this piece of information the evaluator can check if the user successfully finished a task or failed.

In the field of commercial AUT tools, remoterear.ch⁴ gives a good overview of existing products. Two examples for capture and replay are Morae⁵ and Silverback2.0⁶. Furthermore, Localytics⁷, Heatmaps⁸, and Google Analytics⁹ support capture and automated analysis functionality. They include statistics and heatmap visualizations to represent the collection of interaction data. A disadvantage of all tools is the manual integration of capture methods by the developer. None of them supports *automatic critic*.

3 APPROACH

Section 2 gives a good overview of existing approaches and makes clear that no approach exists which uses user interaction data to automatically analyze and critique the usability of mobile applications. In this context, one of the major objectives in this paper is an approach which fulfills this requirement. According to (Baharuddin et al., 2013), the

³<http://graphml.graphdrawing.org/>

⁴<http://remoteresear.ch/tools>

⁵<http://www.techsmith.com/morae.html>

⁶<http://silverbackapp.com>

⁷<http://localytics.com>

⁸<http://heatmaps.io>

⁹<http://www.google.com/analytics/mobile>

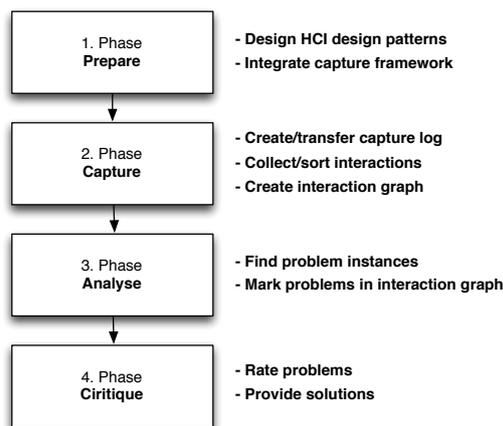


Figure 1: From (Balagtas-Fernandez and Hussmann, 2009) adapted four-phase model for fully AUT tool.

most common approach to evaluate mobile applications is *think aloud* (the user is observed while he is using the application; he talks about what he is actually doing and thinking). What is the method of *think aloud* and how can machines be used to simulate this process to generate similar results? The important steps of this evaluation method is to recognize misuse, usability problems, different and non predicted behavior. The findings are always different because of where they occur. Nevertheless, most of the problems have a significant pattern. Our goal is to determine which problem patterns exist and how they can be found automatically. For this purpose, HCI design patterns (best practices to solve a recurring usability problem)(Borchers, 2000) help to define interaction problems in a usual matter to reuse them for automated analysis and critic. The four-phase model from (Balagtas-Fernandez and Hussmann, 2009) has been adapted with the difference that the phase of extraction and analysis are taken together and a phase of critique has been inserted. The phase of critique allows the developer to get feedback in form of a suggestion for improvement for analyzed usability problems. The purpose of each phase is explained as follows:

- 1. Preparation Phase.** To prepare his project, the developer integrates the AUT capture framework into his application. In addition, he needs HCI design patterns to apply them to the captured interactions. He can reuse patterns from an open platform or he can develop his own patterns.
- 2. Capture Phase.** Each mobile device will automatically generate logs with the user's interaction data. When the session ends, the application will send the log to a central server where it is processed.

- 3. Analysis Phase.** Pattern recognition definition, built from the problem specifications of the HCI design patterns, are applied to each new transmitted log to find and mark related problems.
- 4. Critique Phase.** With the problem HCI design pattern relation a solution in form of best practices for each problem is given. With a higher detail degree of problem specification, the precision of the solution increases.

In this paper a lightweight HCI design pattern definition is used. A pattern consists of a name, a weighting, a problem specification, and a solution. According to the Usability Problem Taxonomy from (Keenan et al., 1999), all considered problems are within the *task-mapping* category consisting of interaction, navigation, and functionality. We assume that especially this kind of usability problems can be found and improved with our approach. The solution to the usability problem, an improvement of the usability, is given in form of best practices based on the HCI design pattern. The weighting scale is based on Nielsen's *severity rating* (Nielsen, 1995).

Hence, we have designed four HCI design patterns presented in Table 1 which satisfy the declared attributes.

The capture component has the purpose of collecting user interactions directly on the mobile device of the user (see Figure 4). An interaction consists of seven attributes: startview, endview, called method, user input-type, timestamp, touch position, and activated UI-element. A log (group of user interactions) is transferred to a server when the user session is over (i.e., when the application is closed). The procedure of capturing is automated and similar to (Lettner and Holzmann, 2012b) and (Weiss and Zduniak, 2007).

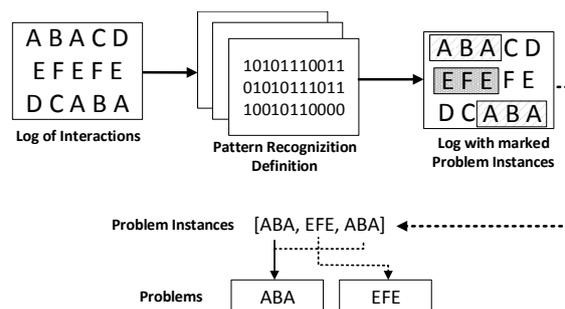


Figure 2: Analyze process with pattern recognition.

In our perspective a usability problem is a sequence of specific user interactions which is defined in the problem specification of a HCI design pattern. E.g., for the pattern *Fitts's Law* all sequences are observed where the user touches repeatedly points next to a button until the button itself is pressed. For this

Table 1: Four HCI design patterns developed for AUT tool.

| Name | Problem Specification | Solution | Weighting | Reference |
|---------------------|--|---|-----------------------------|--|
| Fitts's Law | User misses UI-Element (e.g., button) several times | Make UI-Element bigger and/or move to center | Major usability problem (3) | (Fitts, 1992) & (Henze and Boll, 2011) |
| Silent Misentry | User repeatedly touches UI-elements without functionality (e.g., imageview) | Analyze pressed UI-element and figure out which functionality the user intended; e.g., imageview: image-zoom; add functionality or make clear function is missing | Cosmetic problem only (1) | - |
| Navigational Burden | User switches back and forth between two views multiple times (e.g., master-/detailview) | User is looking for some information which is presented in detailview; needs the way over the masterview to open a new detailview | Minor usability problem (2) | (Ahmad et al., 2006) |
| Accidental Touch | User touches the screen accidentally and activates view change; he immediately revokes input | Check accidentally touched UI-element; move/resize/remove it | Cosmetic problem only (1) | (Matero and Colley, 2012) |

purpose, we use pattern recognition with the user interaction log as input word and the sequence we are looking for as search word (s. Figure 2). Furthermore, it is important to mention that the recognizer looks for dynamic keywords, because in the last example, the views, and the button can change, but it is still a sequence of the same problem and just another instance. As a consequence, we defined an advanced regular expression with dynamic behavior which generalizes the sequences for each pattern. To recognize the usability problems in the pool of interactions, we built from the problem specification of each HCI design pattern a pattern recognition definition based on a regular expression (RE):

- **Fitts's Law:**
 (A^+B) ; $A := (a, e, a)$; $B := (a, x, b)$
- **Navigational Burden:**
 $(AB)^+$; $A := (a, x, b)$; $B := (b, z, a)$
- **Accidental Touch:**
 (AB) ; $A := (a, x, b)$; $B := (b, z, a)$;
 $\Delta t(A, B) < \tau_0$
- **Silent Misentry:**
 (A^+) ; $A := (a, e, a)$

For the sake of simplicity, we reduced the complexity of the regular expression with a simpler definition of interactions. The interaction A in this case consists of a tuple $(startview \times executedmethod \times endview)$ and e represents no-action. The RE (A^+B) describes a search pattern which starts at least with one interaction A and ends with one B . Additional comparisons, e.g., timespan and distance, have to be done manually.

Each problem instance is part of a problem and each problem is derived from a HCI design pattern

which has a solution for it. The accuracy of a solution depends on the accuracy of a problem specification.

For a better visual communication with the developer, a finite state-machine has been designed where all interactions are represented. Hence, nodes stand for views and edges for user-input with executed method (methodname) or no method (e). In addition, each problem instance is a sequence of interactions and can be marked in the graph to indicate in which state a problem occurs for a better understanding (s. Figure 3). As a result, the developer knows immediately in which states a usability problem occurs, the kind of usability problem, and how it could be solved.

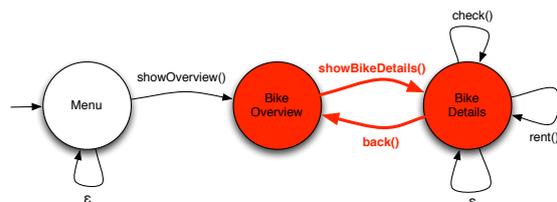


Figure 3: Example for an interaction graph with marked *Navigational Burden* problem.

Figure 4 represents the composition of all phases in one architecture. There are two separated workers, the capturing framework on the client's mobile phone and the central computer unit which evaluates the collected information. Furthermore, the server presents the results of the evaluation in form of a dashboard to the developer.

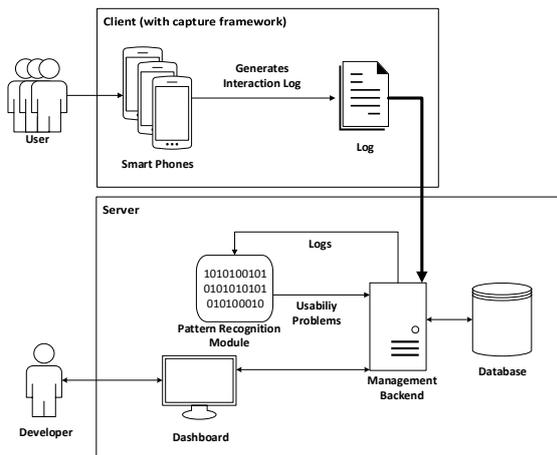


Figure 4: Architecture of the AUT tool.

4 IMPLEMENTATION

Figure 5 shows an overview of all used components for the implementation of the AUT tool. The capture framework has been implemented for Apple’s iOS7 environment, uses *method swizzling*¹⁰ and a *gesture recognizer* (Apple, 2013) to identify user interactions.

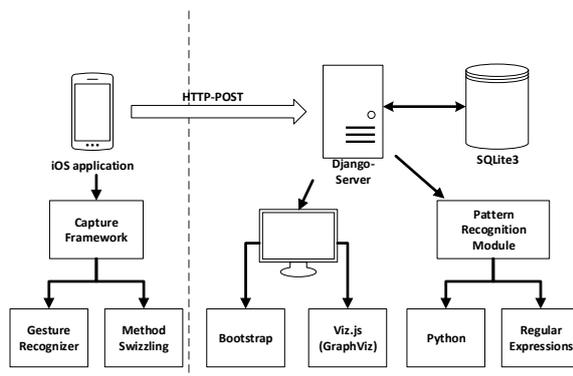


Figure 5: Implementation of the AUT tool.

The log of user interactions is sent via an HTTP-POST to a Python Django 1.5¹¹ webserver. Each capture log is processed in three steps:

1. Loading the interactions into a database
2. Searching for problem instances with pattern recognition modules in collection of interactions
3. Mapping problem instances into suitable problem categories related to their HCI design pattern

The visualization of the information is dynamically generated from database and presented on a dash-

¹⁰<http://cocoadev.com/MethodSwizzling>

¹¹<https://www.djangoproject.com/>

board for the evaluator/developer. The interaction graph is visualized with the help of Viz.js¹² (a GraphViz¹³ implementation in JavaScript) and usability problems are marked in red.

4.1 Capture Framework

For the implementation of the capture framework for iOS7 we had to define the interaction attributes and the technique to gather them. Table 2 shows an overview of the type, name, and technique of an interaction attribute. In our context, an advantage of the iOS environment is the Objective-C runtime which works with messages instead of direct method calls when executing a specific method in the application.

Table 2: Interaction Attributes with Type and used Technique.

| Type | Name | Technique |
|----------------|---------------------|-----------|
| NSNumber* | id | GR |
| BOOL | hasMethod | MS |
| NSString* | methodName | MS |
| NSString* | viewControllerTitle | MS |
| NSString* | viewControllerClass | MS |
| NSTimeInterval | timestamp | GR |
| CGPoint | position | GR |
| NSString* | uiElement | GR |

GR = Gesture Recognizer; MS = Method Swizzling

With *method swizzling* the target method is replaced by another method. This procedure allows us to extend methods of private classes which are responsible for the lifecycle of a view controller and the execution of events with capture functionality (s. Table 3). For this purpose, *categories* (Objective-C; extend existing classes with methods) are used to add a new method with capture functionality to the private class and to call the intended method afterwards. On the other hand, a *gesture recognizer* (UIGestureRecognizer) which automatically identifies a user tap (a single touch event) is added to the root view controller of an application (i.e., foreground view in size of the screen). This allows to recognize when a touch begins and ends in all views. In each start/end callback, the touch event (instance of UIEvent) is included. Table 4 shows the attributes which are set in the delegate methods *touchesBegan* and *touchesEnded*.

Single interaction objects are collected in an array which is sent to the backend with a standard HTTP-POST in JSON format when the application is sent to background.

¹²<https://github.com/mdaines/viz.js>

¹³<http://www.graphviz.org/>

Table 3: Overview of extended private classes.

| Category | Method | Attribute |
|------------------|---------------|-----------------|
| NSObject | responds | hasMethod |
| +Swizzle.h | ToSelector | |
| UIApplication | sendAction | methodName |
| +Swizzle.h | | |
| UIViewController | viewDidAppear | viewController- |
| +Swizzle.h | | Title |
| UIViewController | viewDidAppear | viewController- |
| +Swizzle.h | | Class |

Table 4: Attributes set in Gesture Recognizer Delegate Methods.

| Delegate Method | Attribute |
|-----------------|-----------------------------|
| touchesBegan | <i>init new interaction</i> |
| touchesBegan | id |
| touchesEnded | position |
| touchesEnded | timestamp |
| touchesEnded | uiElement |

4.2 Pattern Recognition Module

Pattern recognition is implemented in Python¹⁴ and has the purpose of finding problem instances in list interactions. Each module implements the method `find_problems_for_pattern(interactions)` which gets a list of interactions as input and returns a list of problem instances. The detailed implementation of the module is provided by the developer. He can use regular expressions, python code or other tools, to describe the target pattern. We designed two example patterns with RE and two programmatically. For the *Fitts's Law* module interactions are converted into string tuples (start-view, executed method, end-view) and the following RE is applied:

```
(\[ (?P<view>\w+), no_action, (?P=view)\], )+
(\[ (?P=view), \w+:?, \w+\])
```

4.3 Management Backend

The backend manages all incoming data. The backend tasks are maintain database, persist user interactions, integration of pattern recognition modules, and visualize results in an interaction graph.

4.4 Data Visualization

The database stores the interactions of all users. To visualize this information in an interaction graph, the attributes of all interactions and problems are reduced to start-view, executed method, and end-view. This

¹⁴<http://www.python.org/>

makes it possible to remove doubled interactions. Afterwards, the structure of the graph is generated as a DOT¹⁵ defined string which can be embedded into the dashboard for the developer. On client side, the Viz.js framework, a GraphViz implementation in JavaScript, transforms the DOT definition in a perceptible interaction graph (s. Figure 6). Recognized usability problems are marked as red.

5 EVALUATION

We compare and evaluate the traditional *user observing* evaluation method in contrast to our AUT tool with a bike sharing prototype application. The application is designed to find problems defined in the four HCI design patterns in Section 3. The flowchart in Figure 7 shows the views and connections of the application. The image with the title “Fahrrad-Verleih” (1st view) provokes a *Silent Misentry*, the lent status in the detailview (3rd and 4th view) causes *Navigational Burden* and the small checkbox in the detailview engenders *Fitts's Law*. The application is developed for iOS7 and the capture framework is integrated and initialized. While the AUT tool generates interaction logs automatically, we had to design a questionnaire for the *User Observing* method which takes care of the user’s behavior considering the usability problems in the four HCI design patterns. Eight participants aged between 23 and 28 years, two researchers and six computer science students attend the evaluation. All of them has experience with smart phones, half with Android and half with iOS.

The participants got the target to rent a bike with this application. The task specifies a context for this evaluation, but it is not necessary for the AUT tool. In our eyes it is still an informal test in the lab and the user can use the app as a normal bike sharing app. The results in Table 5 show that the AUT tool and the classical *user observing* identify all seeded usability problems. In contrast to an automated capturing solution, we had the feeling that *user observing* makes it difficult to capture all user inputs on a touch screen interface. Getting a clear analysis of the *user observing* data was tedious, because we manually sorted it into the classification of interaction design patterns. However, the AUT tool gave us an overview of all problems for each pattern with an appropriate interaction graph.

¹⁵<http://www.graphviz.org/content/dot-language>

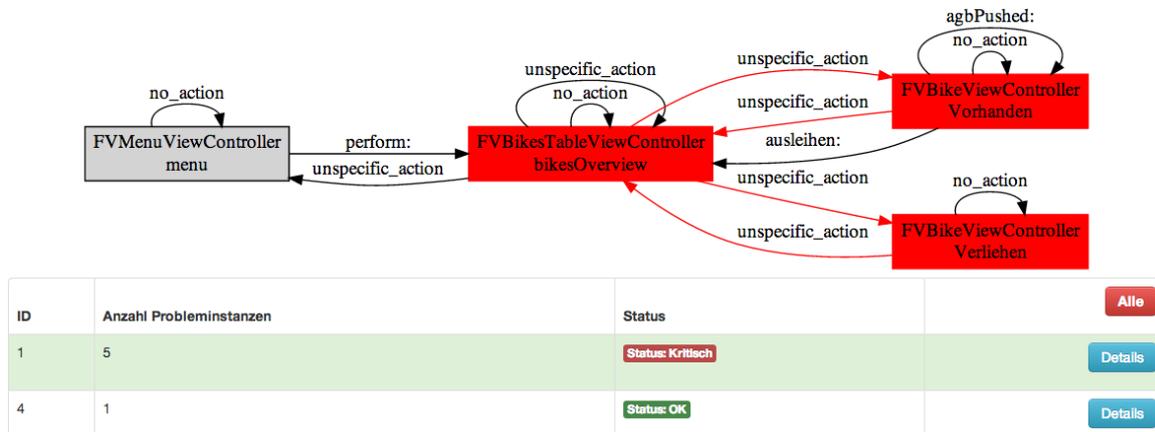


Figure 6: Screenshot of the dashboard visualization for *Navigational Burden*.

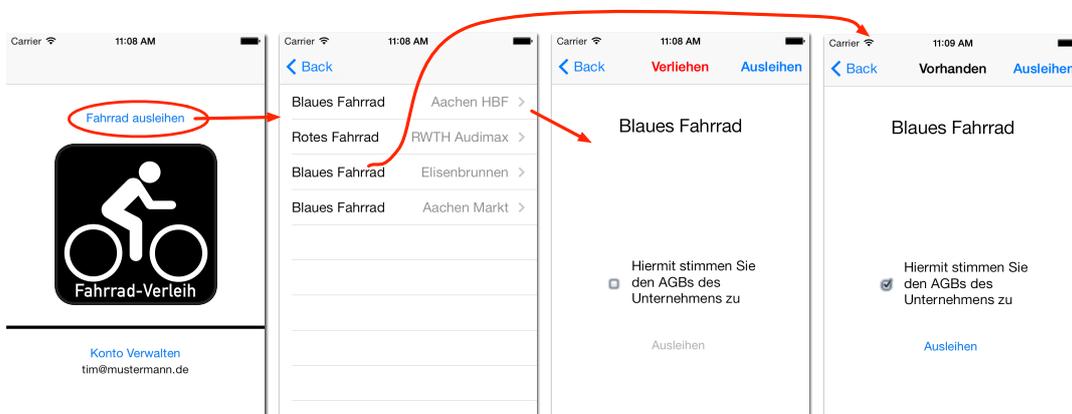


Figure 7: Bike sharing application for iOS7 prepared with usability problems.

Table 5: Results of Bike Sharing App evaluation with *User Observation* and AUT tool.

| HCI design pattern | A | B | C |
|---------------------|---|----|----|
| Fitts's Law | 2 | 9 | 26 |
| Accidental Touch | 5 | 32 | - |
| Silent Misentry | 1 | 4 | 11 |
| Navigational Burden | 2 | 6 | 17 |

A = number of problems (AUT tool)
 B = number of problem instances (AUT tool)
 C = number of problems (user observing)

6 CONCLUSIONS AND FUTURE WORK

Today, usability is one of the major topics in software development and the relevance for mobile applications is still increasing. Automatic testing, e.g., unit tests, shows that it has a huge influence on the software process and quality. With automated usability testing (AUT) we believe that usability testing will

become a firm part of the software development process.

The objective was to develop an automated usability testing tool for iOS applications to get a cheaper, faster and simpler usability evaluation process. We found no existing approaches which fulfill the criteria of automatic critic for mobile applications which takes the responsibility of usability decisions from the developer to a tool.

For this purpose we modified the four phase model from (Balagtas-Fernandez and Hussmann, 2009) and extended it with a phase of *automatic critic*. Usability problems are recognized by pattern recognition which works with definitions from four exemplary HCI design patterns. We implemented a fully automated capture framework for iOS applications and a backend for data management, analysis and representation.

We evaluated the AUT tool using a prototype bike sharing application and compared it to a classic evaluation method, *user observing*. The results show that the AUT tool is able to find all usability problems which are described in the interaction design patterns.

In contrast to *user observing*, working with the AUT tool was less complicated and time-consuming for the developer, which, for us, is an evidence for success.

Future Work

In the next version of the AUT tool, we are going to improve the visualization of the graph by showing a screenshot of the current view as node and the edge will begin directly from the point where the user touched (similar to Figure 7). In addition, we will create more HCI design patterns and a tool which makes the design process much simpler. We also plan to integrate existing fully automated capture frameworks for other mobile platforms (s. Section 2.1).

ACKNOWLEDGEMENTS

We would like to thank our former colleague Paul Heiniz for his support in the early phase of this project. This work was supported by the German Federal Ministry of Economics and Technology¹⁶: **(Grant 01ME12052 econnect Germany, Grant 01ME12136 Mobility Broker).**

REFERENCES

- Ahmad, R., Li, Z., and Azam, F. (2006). Measuring navigational burden. In *Software Engineering Research, Management and Applications, 2006. Fourth International Conference on*, pages 307–314.
- Albraheem, L. and Alnuem, M. (2012). Automated Usability Testing : A Literature Review and an Evaluation.
- Apple (2013). iOS Developer Library: UIGestureRecognizer Class Reference.
- Au, F., Baker, S., Warren, I., and Dobbie, G. (2008). Automated Usability Testing Framework. volume 76, pages 55–64.
- Baharuddin, R., Singh, D., and Razali, R. (2013). Usability Dimensions for Mobile Applications-A Review. *Research Journal of Applied Sciences, Engineering and Technology*, 5(6):2225–2231.
- Baker, S. and Au, F. (2008). Automated Usability Testing Using HUI Analyzer. pages 579–588.
- Balagtas-Fernandez, F. and Hussmann, H. (2009). A Methodology and Framework to Simplify Usability Analysis of Mobile Applications. pages 520–524. IEEE.
- Borchers, J. O. (2000). A pattern approach to interaction design. pages 369–378.
- Fitts, P. M. (1992). The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement. *Journal of Experimental Psychology: General*, 121(3):262–9.
- Henze, N. and Boll, S. (2011). It Does Not Fitts My Data! Analysing Large Amounts of Mobile Touch Data Niels. *INTERACT 2011, Part IV*, pages 564–567.
- Ivory, M. and Hearst, M. (2001). The state of the art in automating usability evaluation of user interfaces. *ACM Computing Surveys*, 33(4):470–516.
- Keenan, S. L., Hartson, H. R., Kafura, D. G., and Schulman, R. S. (1999). The Usability Problem Taxonomy: A Framework for Classification and Analysis. *Empirical Software Engineering*, 4:71–104.
- Lettner, F. and Holzmann, C. (2012a). Sensing mobile phone interaction in the field. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on*, pages 877–882.
- Lettner, F. and Holzmann, C. (2012b). Usability evaluation framework: Automated interface analysis for android applications. In *Proceedings of the 13th International Conference on Computer Aided Systems Theory - Volume Part II, EUROCAST'11*, pages 560–567, Berlin, Heidelberg. Springer-Verlag.
- Matero, J. and Colley, A. (2012). Identifying unintentional touches on handheld touch screen devices. pages 506–509.
- Nielsen, J. (1995). Severity Ratings for Usability Problems.
- Schmidt, A. (2000). Implicit human computer interaction through context. *Personal and Ubiquitous Computing*, 4(2-3):191–199.
- Weiss, D. and Zduniak, M. (2007). Automated integration tests for mobile applications in java 2 micro edition. pages 478–487.
- Zhang, D. and Adipat, B. (2005). Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications. *International Journal of Human-Computer*, pages 293–308.

¹⁶Bundesministerium für Wirtschaft und Technologie (BMWi) <http://www.bmwi.de/>

WEB INTELLIGENCE

FULL PAPERS

The GENIE Project

A Semantic Pipeline for Automatic Document Categorisation

Angel L. Garrido, Maria G. Buey, Sandra Escudero, Alvaro Peiro, Sergio Ilarri and Eduardo Mena

IIS Department, University of Zaragoza, Zaragoza, Spain
{garrido, mgbuey, sandra.escudero, peiro, silarri, emena}@unizar.es

Keywords: Knowledge Management, Text Mining, Ontologies, Linguistics.

Abstract: Automatic text categorisation systems is a type of software that every day it is receiving more interest, due not only to its use in documentaries environments but also to its possible application to tag properly documents on the Web. Many options have been proposed to face this subject using statistical approaches, natural language processing tools, ontologies and lexical databases. Nevertheless, there have been no too many empirical evaluations comparing the influence of the different tools used to solve these problems, particularly in a multilingual environment. In this paper we propose a multi-language rule-based pipeline system for automatic document categorisation and we compare empirically the results of applying techniques that rely on statistics and supervised learning with the results of applying the same techniques but with the support of smarter tools based on language semantics and ontologies, using for this purpose several corpora of documents. GENIE is being applied to real environments, which shows the potential of the proposal.

1 INTRODUCTION

In almost any public or private organization that manages a considerable amount of information, activities related to text categorisation and document tagging can be found. To do this job, large organizations have documentation departments. However, the big amount of information in text format that organizations usually accumulate cannot be properly processed and documented by these departments. Besides, the manual labor of labeling carried out by these people is subject to errors due to the subjectivity of the individuals. That is why a tool that automates categorisation tasks would be very useful, and would help to improve the quality of searches that are performed later over the data.

To perform these tasks, software based on statistics and the frequency of use of words can be used, and it is also very common to use machine learning systems (Sebastiani, 2002). However, we think that other kinds of tools capable of dealing with aspects related to Natural Language Processing (NLP) (Smeaton, 1999) are also necessary to complement and enhance the results provided by these aforementioned techniques.

Moreover, to perform any task related to the processing of text documents, it is highly recommended to own the know-how of the organization, so it is

highly advisable to manage ontologies (Gruber et al., 1993) and semantic tools such as reasoners (Mishra and Kumar, 2011) to make knowledge explicit and reason over it, respectively. Furthermore, it is very common for organizations to have their own catalog of labels, known as thesaurus (Gilchrist, 2003), so it is important that the system is able to obtain not only keywords from the text, but also know how to relate them to the set of thesaurus descriptors.

Furthermore, this same issue is also found in the Web where much of the information is in text format. Providing tools capable of automatically tagging web pages is something very helpful in order to improve the search and retrieval tasks of information using search engines, and today is still an open problem (Atkinson and Van der Goot, 2009; Chau and Yeh, 2004) due (among others) to the existing semantic and linguistic difficulties to process.

Our purpose is to bring together these techniques into an architecture that enables the automatic classification of texts, with the particular feature that it exploits different semantic methods, which is added as a new element in text categorization to support typical techniques that rely on statistics and supervised learning. Although there are some researches in text categorization that takes into account Spanish texts as examples, there are no tools especially focused on the Spanish language. Moreover, the proposed sys-

tem has been implemented to be open to allow the possibility to add the analysis of other languages, like English, French, or Portuguese.

Other important characteristics of the architecture is that it has been proposed as a pipeline system and it has been implemented with different modules. We consider these as important features because a pipeline system gives us the chance to control the results at each phase of the process and also the structure with different modules allows us to easily upgrade its individual components. For example, geographic or lexical databases change over time, and our modular architecture easily accommodates these changes.

The fact that the system is implemented in different modules is also interesting because it is ideal when performing the analysis of a text. Sometimes, we may want not to have to use all the modules that make up the architecture to achieve a desired result. For example, we may want to extract only statistical information from the words present in a text, but nothing related about their semantics. Also, it is possible that we need to change the order of the modules a text passes through depending on the type of analysis of the text we want to perform. For these reasons it is important to consider a modular architecture: it makes the system easy to use and it facilitates improving it over time.

This paper provides two main contributions:

- Firstly, we present a tool called GENIE, whose general architecture is valid for text categorisation tasks in any language. This system has been installed and tested in several real environments using different datasets. The set-up of our algorithm is rule-based and we use for inference the document's features as well as the linguistic content of the text and its meaning.
- Secondly, we experimentally quantify the influence of using linguistic and semantic tools when performing the automatic classification, working on a real case with Spanish texts previously classified by a professional documentation department.

The rest of this paper is structured as follows. Section 2 explains the general architecture of the proposed categorisation system. Section 3 discusses the results of our experiments with real data. Section 4 analyzes other related works. Finally, Section 5 provides our conclusions and future work.

2 PROPOSED ARCHITECTURE

In this section, we explain the general architecture of the proposed system as well as and the corresponding

working methodology. The system relies on the existence of several resources. First, we will describe these resources, and then we will explain in detail the classification process (see Figure 1).

2.1 Resources

Regarding resources, we have to consider both static data repositories and software tools:

- *Thesaurus*. A thesaurus is a list of words and a set of relations among them, used to classify items (Gilchrist, 2003). We use its elements as the set of tags that must be used to categorize the set of documents. Examples of thesaurus entries are words like HEALTH, ACCIDENT, FOOTBALL, BASKETBALL, REALMADRID, CINEMA, JACK_NICHOLSON, THEATER, etc. The terms can be related. For example, FOOTBALL and BASKETBALL could depend hierarchically on SPORTS. Each document may take a variable number of terms in the thesaurus during the categorisation process.
- *Gazetter*. A gazetteer is a geographic directory containing information about places and place names (Goodchild and Hill, 2008). In our system, it is used to identify geographic features.
- *Morphological Analyzer*. It is an NLP tool whose mission is the identification, analysis and description of the structure of a set of given linguistic units. This analyzer consists of a set of different analysis libraries, which can be configured and used depending on the working language, and a custom middleware architecture which aims to store all the different analysis results in structures that represent the desired linguistic units, such as words (with their morphological and lexical information), sentences (with their syntax and dependency trees) and texts. With this approach we can provide the same entities to the other modules that work with NLP, resulting in an architecture that can work with multiple analysis tools and languages.
- *Lexical Database*. A lexical database is a lexical resource which groups words into sets of synonyms called *synsets*, includes semantic relations among them, and provides definitions. Examples could be *WordNet* (Miller, 1995) and *EurowordNet* (Vossen, 1998).
- *Stop Word List*. This is a list of frequent words that do not contain relevant semantic information (Wilbur and Sirotkin, 1992). In this set we may include the following types of words: articles, conjunctions, numbers, etc.

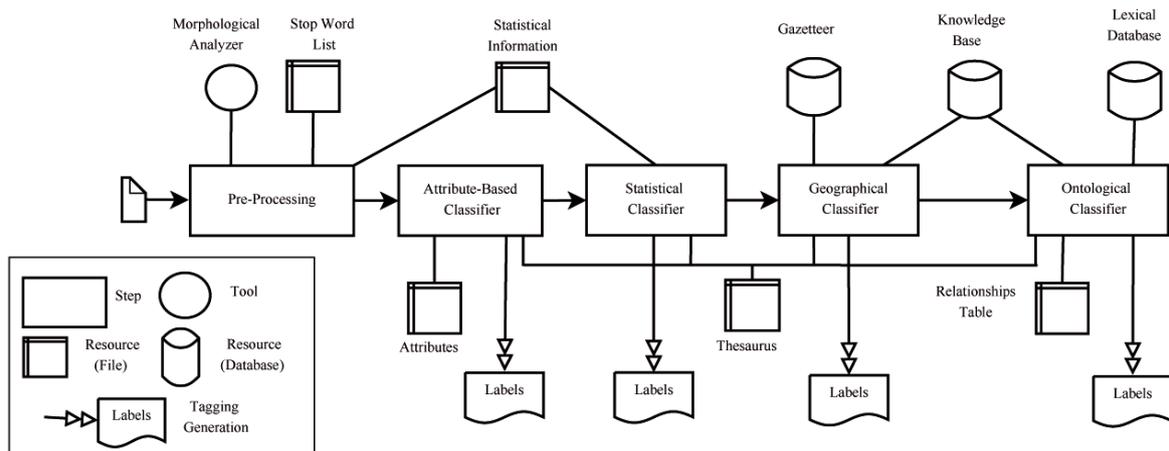


Figure 1: General pipeline of GENIE, the proposed text categorisation system.

- *Knowledge Base*. This refers to the explicit representation of knowledge related to the topics covered in the documents that have to be catalogued. As a tool for knowledge representation in a software system, we use ontologies. An ontology is a formal and explicit specification of a shared conceptualization (Gruber et al., 1993) that provides a vocabulary of classes and relations to describe a particular area, supporting automatic inferences by using a reasoner (Mishra and Kumar, 2011). The idea is to represent in these ontologies the concepts that could help to label a document in a given context, and to populate the ontologies with as many instances as possible.
- *Statistical Information*. This consists of a set of files with information about the use frequency of each word, related to the attributes of the text and to the set of elements in the thesaurus. For example: the word “ONU” appears more frequently in documents of type “International” and it is related with the descriptor INTERNAT in a thesaurus used in the documentation department of a newspaper we have worked with. These frequencies allow us to estimate if a given text can be categorized with a particular element of the thesaurus.
- *Relationships Table*. This table relates items in the Gazetteer and knowledge base concepts with thesaurus elements. It may be necessary in an organization because the concepts stored in the semantic resources available may not match the labels in the thesaurus that must be considered for classification. The construction of this table could be manual or automatic, using any machine learning method.

As we will show in the experimental evaluation, the use of some resources is optional, leading to dif-

ferent results in terms of the expected performance of the system. This system could be used with different languages by changing the language-dependent resources, i.e. the Gazetteer, the NLP tool, the lexical database, and the stop word list.

2.2 Process Pipeline

We have used a pipeline scheme with separated stages. Each of the stages is associated with only one type of process and they communicate between themselves through different files. Although it is a pipeline system, the process can be configured so that each of the tasks can be activated or deactivated depending on whether we want the text document to go through certain phases or not. For example, we may want to use the pipeline without having activated the *Geographical Classifier*. This choice has three purposes:

1. The early stages perform a more general classification, and later phases make more specific labeling that requires more precise resources. We have verified, through experimental evaluation, that taking advantage of a filter to select the most appropriate resources for the later stages improves the results.
2. Separating each stage simplifies control for evaluation. We know that there are certain tasks that could be parallelized, but the aim is to analyze the results in the best possible way, rather than to provide an optimized algorithm.
3. We have more freedom to add, delete or modify any of the stages of the pipeline if they are independent. If we would like to use a different tool in any of the stages, changing it is very easy when there is a minimum coupling between phases.

Our system works over a set of text documents, but we have to note that each of them could have a variable number of attributes (author, title, subtitle, domain, date, section, type, extension, etc.), that we will use during the categorisation process. These attributes vary according to the origin of the document: a digital library, a database, a website, etc. Numeric fields, dates, strings, or even HTML tags may be perfectly valid attributes to the system. We could also consider as attributes those elements that are specific to the structure of the type of document, such as hyperlinks (Shen et al., 2006) in the case of web pages. As a very first stage, the system includes specific interfaces to convert the original documents into XML files with a specific field for plain text and others for attributes.

The tasks for the proposed automatic text categorisation system are:

1. *Preprocessing* of the text of the document, which consists of three steps:
 - (a) *Lemmatization*. Through this process we obtain a new text consisting of a set of words corresponding to the lemmas (canonical forms) of the words in the initial text. This process eliminates prepositions, articles, conjunctions and other words included in the *Stop Words List*. All the word information (Part of Speech, gender, number) is stored in the corresponding structure, so it can be recovered later for future uses.
 - (b) *Named Entities Recognition (NER)*. Named entities are atomic elements in a text representing, for example, names of persons, organizations, locations, quantities, monetary values, or percentages (Sekine and Ranchhod, 2009). By using a named entity extractor, this procedure gets a list of items identified as named entities. This extractor can be paired with a statistical Named Entity Classification (NEC) in a first attempt to classify the named entity into a pre-defined group (person, place, organization) or leave it undefined so the following tasks (Geographical Classifier) can disambiguate it.
 - (c) *Keywords Extraction*. Keywords are words selected from the text that are in fact key elements to consider to categorize the document. We use the lemmatized form of such words and the TF/IDF algorithm (Salton and Buckley, 1988).

These processes produce several results that are used in subsequent stages. The resources used in this stage are the morphological analyzer, the Stop Word List and the statistical data.
2. *Attributes-Based Classifier*. Taking advantage of the attributes of each of the documents, this ruled-based process makes a first basic and general tagging. For example, if we find the words “film review” in the “title” field the system will infer that the thesaurus descriptor CINEMA could be assigned to this document. At the same time, it establishes the values of the attributes to be used for the selection of appropriate resources in the following steps, choosing for instance an ontology about cinema for the Ontological Classifier stage.
3. *Statistical Classifier*. Using machine learning techniques (Sebastiani, 2002), the document text is analyzed to try to find patterns that correspond to data in the files storing statistical information. This step is mainly useful to try to obtain labels that correspond to the general themes of the document. Trying to deduce if a document is talking about football or basketball could be a good example.
4. *Geographical Classifier*. By using the gazetteer, named entities (NE) corresponding to geographical locations are detected. This stage is managed by a ruled-based system. Besides, it can deal with typical disambiguation problems among locations of the same name and locations whose names match other NE (e.g., people), by using the well-known techniques described in (Amitay et al., 2004): usually there is only single sense per discourse (so, an ambiguous term is likely to mean only one of its senses when it is used multiple times), and place names appearing in the same context tend to show close locations. Other important considerations that GENIE takes into account are to look at the population of the location candidates as an important aspect to disambiguate places (Amitay et al., 2004) and consider the context where the text is framed to establish a list of bonuses for certain regions (Quercini et al., 2010). Other used techniques are to construct an N-term window on both sides of the entity considered to be a geographic term, as some words can contribute with a positive or negative modifier (Rauch et al., 2003), or to try to find syntactic structures like “city, country” (e.g. “Madrid, Spain”) (Li et al., 2002). Finally, using techniques explained in (Garrido et al., 2013b), the system uses ontologies in order to capture information about important aspects related to certain locations. For example: most important streets, monuments and outstanding buildings, neighborhoods, etc. This is useful when a text has not explicit location identified. Besides, it takes advantage too of the results of previous stages. For example, if in the previ-

ous stages we got the descriptor EUROPE we can assign higher scores to the results related to European countries and major European cities than to results related to locations in other continents. The geographical tagging unit is very useful because, empirically, near 30% of tags in our experimental context are related to locations.

5. *Ontological Classifier.* To perform a detailed labeling, the knowledge base is queried about the named entities and keywords found in the text. If a positive response is obtained, it means that the main related concepts can be used to label the text. A great advantage is that these concepts need not appear explicitly in the text, as they may be inferred from existing ontological relations. If there is an ambiguous word, it can be disambiguated (Resnik, 1999) by using the Lexical Database resource (for a survey on word sense disambiguation, see (Navigli, 2009)). As soon as a concept related to the text is found, the relations stored in the *Relationships Table* are considered to obtain appropriate tags from the thesaurus. As explained before, the fact that at this phase we have a partially classified document allows us to choose the most appropriate ontologies for classification using configurable rules. For example, if we have already realised with the statistical classifier that the text speaks of the American Basketball League, we will use a specific ontology to classify the document more accurately finding out for instance the teams and the players, and we will not try to use any other resource. This particular ontology could be obtained and re-used from the Web. But if we had discovered that the text is about local politics, we will use another specific ontology to deduce the most appropriate tags. This ontology would probably be hand-made, or it would be adapted from other similar ontology, because this kind of resources are difficult or impossible to find for free on the Web. So, our system is generic enough to accommodate the required and more appropriate ontologies (existing or hand-made) for the different topics covered in the texts.

The way to obtain the tags is asking about keywords and NE to the ontology, by using SPARQL¹, a set of rules, and the relationship table to deduce the most suitable tags. The behavior of the ontology is not only to be a simple *bag-of-words*, because it can contain concepts, relations and axioms, all of them very useful to inquire the

implicit topics in the text.

In summary, the text categorization process that GENIE performs consists of following each of the proposed tasks that constitute the system's pipeline. This process begins with the preprocessing of the input text, which implies labors of lemmatization of the text and extraction of named entities and keywords from the text. Then it analyzes a set of attributes that are given with the text that is being analyzed in order to extract the first basic and general labels. Afterwards, it applies a statistical classification method based on machine learning techniques to obtain labels that correspond to the general themes of the document. Then it applies a geographic classifier for the purpose of identifying possible geographical references included in the text. Finally, it applies an ontological classifier in order to carry out a more detailed classification of the text, which performs an analysis of named entities and keywords obtained from the text, consults the appropriate ontology, and uses a lexical database to remove possible ambiguities.

3 EXPERIMENTAL EVALUATION

We have performed a set of experiments to test and compare the performance of our architecture with others tools. For this purpose, we have tested in a real environment using three corpus of news previously labeled by a professional documentation department of several major Spanish Media: *Heraldo de Aragón*², *Diario de Navarra*³ and *Heraldo de Soria*⁴. Each corpus had respectively 11,275, 10,200, and 4,500 news. These corpora are divided in several categories: local, national, international, sports, and culture. Every media has a different professional thesaurus used to classify documents, with more than 10,000 entries each. For classification, each document can receive any number of descriptors belonging to the thesaurus. The ideal situation would be that the automatic text categorization system could perform a work identical to the one performed by the real documentation departments.

These news are stored in several databases, in tables where different fields are used to store the different attributes explained in Section 2 (title, author, date, section, type, extension, etc.). For experimental evaluation, we have extracted them from the databases and we have put each text and the data of its fields in XML files. We have used this corpus of XML

¹<http://www.w3.org/TR/2006/WD-rdf-sparql-query-20061004/>

²<http://www.heraldo.es/>

³<http://www.diariodenavarra.es/>

⁴<http://www.heraldodesoria.es/>



Figure 2: GENIE control interface.

files as the input of the system, and the output is the same set of files but with an additional field: classification information. This new XML node contains the set of words (descriptors) belonging to the thesaurus used to categorize the document, i.e., this node contains the different tags that describe the XML file. An example of node that can be contained in an XML file of cinema is:

```
<classify>
CULTURE CINEMA WOODY_ALLEN
</classify>
```

As the news in the dataset considered had been previously manually annotated by the professionals working in the documentation department, we can compare the automatic categorization with that performed by humans. So, we can evaluate the number of hits, omissions and misses.

3.1 Experimental Settings

In the experiments, we have examined the following measures, commonly used in the Information Retrieval context (Manning et al., 2008): the *precision*, the *recall*, and the *F-Measure*. The dataset used initially in the experiments has been the Heraldo de Aragón corpus. We have used the information from this dataset to define most of the rules of the various processes associated with each of the stages of the classification system. These rules are integrated in a configuration file which contains all the information necessary to lead the process and obtain the correct result. The other two datasets (Diario de Navarra and Heraldo de Soria) have been used just to double-check if the application of those rules also produced the desired result; for comparison purposes, at the end of this section we will also present some experimental results based on them. Since news, thesauri, ontologies and classification fields are private data of each company, they are not available on-line on the Web⁵.

⁵Anyway, if any researcher wants to use our corpus for experimental purposes, he/she is entitled to apply directly to the first author and they will be provided privately.

In a first stage, we have used *MALLET* (McCallum, 2002) to classify the different news corpus. *MALLET* is a tool that allows the classification of documents. A classifier is an algorithm that distinguishes between a fixed set of classes, such as “spam” vs. “non-spam”, based on labeled training examples. *MALLET* includes implementations of several classification algorithms, including Naive Bayes, Maximum Entropy, and Decision Trees. In addition, *MALLET* provides tools for evaluating classifiers. In addition to classification, *MALLET* includes tools for sequence tagging for applications such as the extraction of named entities from text. The algorithms include Hidden Markov Models, Maximum Entropy Markov Models, and Conditional Random Fields. These methods are implemented in an extensible system for finite state transducers. The following classifiers were used: MaxEnt, Naive Bayes, C45 and DecisionTree, achieving in the best case 60% in all of the three measures (precision, recall and F-measure).

Afterwards, we have performed four experiments with our own classifier. The appearance of the GENIE control application can be seen in Figure 2. Each stage of the pipeline can be enabled or disabled separately. Regarding the resources and tools considered, we have used Freeling (Carreras et al., 2004), as the Morphological Analyzer and Support Vector Machines (SVM) (Joachims, 1998) to automatically classify topics in the Statistical Classifier.

We have chosen Freeling as is the only and widely used active analysis tool suite that supports several analysis services in both Spanish and English, as well other languages which can be incorporated in our architecture in future developments. Some implementation details of Freeling were modified in order to encapsulate it as a consistent library, incorporating it into our architecture. As Freeling outputs their analysis results in an undesired format to our approach, the need to construct new structures for the several linguistic units was necessary to define an architecture which can support this library and other different analysis tools than can be added in the future. These structures aim to group most of the mutual characteristics of the Romanic languages and the English language into a single approach, while singular language features had to be handled apart.

Regarding the type of SVM used, we have used a modified version of the Cornell SVM-Light implementation (Joachims, 2004) with a Gaussian radial basis func-

tion kernel and the term frequency of the keywords as features (Leopold and Kindermann, 2002). To obtain the frequencies we have used a different corpus of 100,000 news, in order to get a realistic frequency information. Finally, we have chosen Eurowordnet as the Lexical Database and *Geonames*⁶ as the Gazetteer.

To train this Statistical Classifier we have used sets of 5,000 news for each general theme associated to one descriptor (FOOTBALL, BASKET, CINEMA, HANDBALL, and MUSIC). These sets of news are different from the datasets used in the experiments (as is obviously expected in a training phase). For each possible descriptor, we have an ontology, in this case we have designed five ontologies using OWL (McGuinness et al., 2004) with near a hundred concepts each one.

Next, there is an example of a piece of news:

This weekend is the best film debut for the movie “In the Valley of Elah”. The story revolves around the murder of a young man who has just returned from the Iraq war, whose parents try to clarify with the help of the police. As interpreters we have: Tommy Lee Jones, Susan Sarandon and Charlize Theron. Writer-director Paul Haggis is the author of “Crash” and writer of “Million Dollar Baby”, among others.

In this case, the system analyzes and classifies the text with the descriptor CINEMA. Moreover, the news can be tagged with tags such as C.THERON, IRAQ, TL.JONES, etc.

3.2 Experimental Results

We have compared our classification of the 11,275 news in the first dataset with the original classification made by professionals. The results can be seen in Figure 3. Below we analyze the experiments:

1. In the first experiment (*Basic*) we have used the process presented in Section 2 without the Pre-Processing step and without the Ontological Classifier. We have trained the system with SVM to classify 100 themes. In this case, as we do not use the steps of Pre-Processing and the Ontological Classifier, the system has not performed the lemmatization, the named entities recognition, the keywords extraction, and the detailed labeling of the text. For this reason, the precision and the recall are not good, as it is essential to embed semantic information and conceptual patterns in order to enhance the prediction capabilities of classification algorithms.
2. In the second one (*Semantic*), we have introduced the complete Pre-Processing stage and its associated resources, we have used the Lexical Database EuroWordNet (Vossen, 1998) to disambiguate keywords, and we have introduced the Ontological Classifier, with five ontologies with about ten concepts and about 20 instances each. In this experiment the precision and the recall slightly improved because, as explained before, the step of Pre-Processing is important to obtain a better classification.

3. In the third one (*Sem + Geo*) we have included the Geographical Classifier but we have used only the Gazetteer resource. Here we have improved the recall of the labeling but in exchange of a decrease in the precision. By analyzing the errors in detail, we observe that the main cause is the presence of misclassifications performed by the Geographical Classifier.
4. Finally, in the fourth experiment (*Full Mode*), we have executed all the pipeline, exploited all the resources and populated the ontologies with about one hundred instances, leading to an increase in both the precision and the recall. Ontology instances added in this experiment have been inferred from the observation of the errors obtained in previous experiments. The motivation to add them is that otherwise the text includes certain entities unknown to the system, and when they were incorporated this helped to improve the classification.

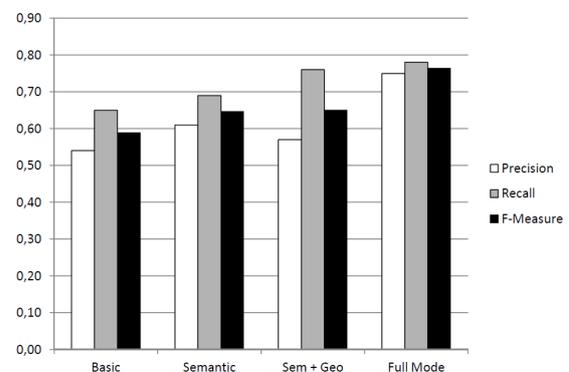


Figure 3: Results of the four document categorisation experiments with news in the dataset 1.

If we look at the overall results obtained in the experiment 1 and the experiment 2 in the Figure 3, we could say that the influence of using semantic and NLP tools is apparently not so significant (about 20%). However, it seems clear that these tools significantly improve the quality of labeling in terms of precision, recall and F-measure, reaching up to about the 80%. Therefore, the use of semantic techniques can make a difference when deciding about the possibility to perform an automatic labeling.

After evaluating the results obtained in the reference dataset (Heraldo de Aragón), we repeated the same experiments with the two other datasets. These dataset were not considered while designing the ontologies, in order to maintain the independence of the tests. The results can be seen in Figure 4. The results obtained with datasets different from the one used for Heraldo de Aragón, which was used to configure the rule-based system, are only slightly different (differences smaller than 10%). In Figure 4, it can also be seen that the trends of the results are very similar regardless of the data. This shows the generality of our approach, since the behavior of the classification system has been reproduced with several different corpora. Experimental results have shown that with our approach, in all the experiments, the system has improved the results achieved by basic machine learning based systems.

⁶<http://www.geonames.org/>

4 RELATED WORK

Text categorisation represents a challenging problem for the data mining and machine learning communities, due to the growing demand for automatic information retrieval systems. Systems that automatically classify text documents into predefined thematic classes, and thereby contextualize information, offer a promising approach to tackle this complexity (Sebastiani, 2002).

Document classification presents difficult challenges due to the sparsity and the high dimensionality of text data, and to the complex semantics of natural language. The traditional document representation is a word-based vector where each dimension is associated with a term of a dictionary containing all the words that appear in the corpus. The value associated to a given term reflects its frequency of occurrence within the corresponding document and within the entire corpus (the *tf-idf* metric). Although this is a representation that is simple and commonly used, it has several limitations. Specifically, this technique has three main drawbacks: (1) it breaks multi-word expressions into independent features; (2) it maps synonymous words into different components; and (3) it considers polysemous words as one single component. While a traditional preprocessing of documents, such as eliminating stop words, pruning rare words, stemming, and normalization, can improve the representation, its effect is also still limited. So, it is essential to embed semantic information and conceptual patterns in order to enhance the prediction capabilities of classification algorithms.

Research has been done to exploit ontologies for content-based categorisation of large corpora of documents. WordNet has been widely used, for example in (Siolas and d'Alché Buc, 2000) or (Elberrichi et al., 2008), but their approaches only use synonyms and hyponyms, fail to handle polysemy, and break multi-word concepts into single terms. Our approach overcomes these limitations by incorporating background knowledge derived from ontologies. This methodology is able to keep multi-word concepts unbroken, it captures the semantic closeness to synonyms, and performs word sense disambiguation for polysemous terms.

For disambiguation tasks we have taken into account an approximation described in (Trillo et al., 2007), that is based on a semantic relatedness computation to detect the set of words that could induce an effective disambiguation. That technique receives an ambiguous keyword and its context words as input and provides a list of possible senses. Other studies show how background knowledge in form of simple ontologies can improve text classification results by directly addressing these problems (Bloehdorn and Hotho, 2006), and others make use of this intelligence to automatically generate tag suggestions based on the semantic content of texts. For example (Lee and Chun, 2007), which extracts keywords and their frequencies, uses WordNet as semantics and an artificial neural network for learning.

Among other related studies that quantify the quality of an automatic labeling performed by using ontologies, we could mention (Maynard et al., 2006; Hovy et al., 2006), but both are focused on a purely semantic labeling (i.e., they do not consider statistics-based methods). More related to our study, it is interesting to mention the work presented in (Scharkow, 2013), although it does not include much in-

formation about the use of ontologies. Examples of hybrid systems using both types of tools include the web service classifier explained in (Bruno et al., 2005), the system *NASS (News Annotation Semantic System)* described in (Garrido et al., 2011; Garrido et al., 2012), which is an automatic annotation tool for the Media, or *GoNTogle* (Bikakis et al., 2010), which is a framework for general document annotation and retrieval.

5 CONCLUSIONS AND FUTURE WORK

A tool for automating categorisation tasks is very useful nowadays, as it helps to improve the quality of searches that are performed later over textual repositories like digital libraries, databases or web pages. For this reason, in this paper we have presented a pipeline architecture to help in the study of the problem of automatic text categorisation using specific vocabulary contained in a thesaurus. Our main contribution is the design of a system that combines statistics, lexical databases, NLP tools, ontologies, and geographical databases. Its stage-based architecture easily allows the use and exchange of different resources and tools. We have also performed a deep study of the impact of the semantics in a text categorisation process.

Our pipeline architecture is based on five stages: preprocessing, attribute-based classification, statistical classification, geographical classification, and ontological classification. Although the experimental part has been developed in Spanish, the tool is ready to work with any other language. Changing linguistic resources suitable for the intended language is enough to make the system work, since the process is sufficiently general to be applicable regardless of the language used. The main contribution of our work is, apart from the useful and modular pipeline architecture, the experimental study with real data of the problem of categorization of natural language documents written in Spanish. There are many studies related to such problems in English, but it is difficult to find them in Spanish. Besides, we have compared the impact of applying techniques that rely on statistics and supervised learning with the results obtained when semantic techniques are also used. There are two remarkable aspects. Firstly, enhancing the amount of knowledge available by increasing the number of instances in the ontologies leads to a substantial improvement in the results. Secondly, the use of knowledge bases helps to correct many errors from a Geographical Classifier.

Spanish vs. English Language. Our research on this topic focuses on transfer projects related to the extraction of information, so for us it is very important to work with real cases. Therefore, the comparison of our work with typical benchmark data sets in English is not fundamental to us, since they are not useful to improve the performance of our system in Spanish, and we have seen that the ambient conditions (language, regional context, thematic news, etc.) have a great influence on the outcome of experiments. Many researchers have already analyzed the differences between working in NLP topics in English and in Spanish, and they

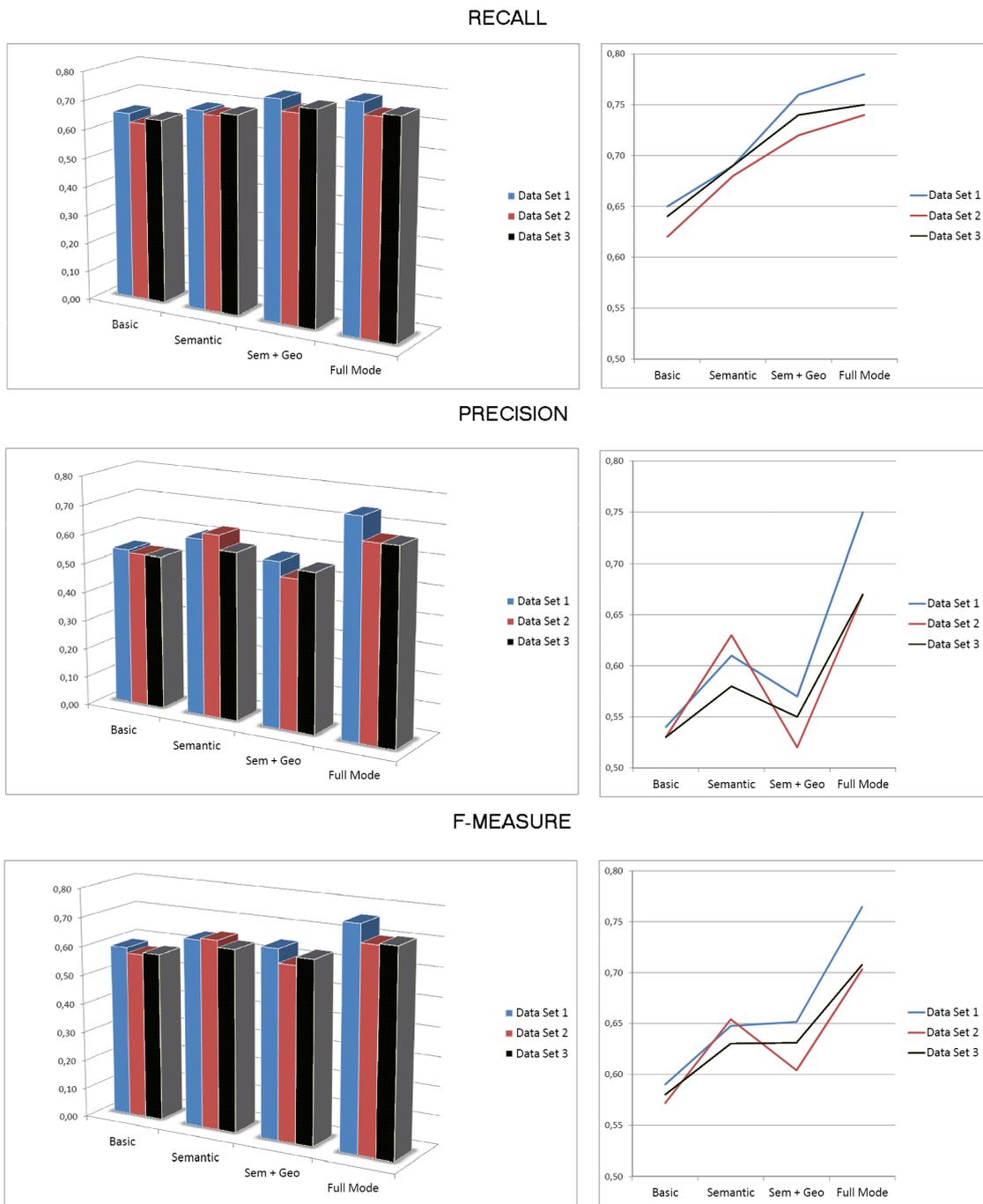


Figure 4: Comparative results of the automatic categorisation experiments.

have made it clear the additional difficulties of the Spanish Language (Carrasco and Gelbukh, 2003; Aguado de Cea et al., 2008), which could explain the poor performance of some software applications that work reasonably well in English. Just to mention some of these differences: in Spanish words contain much more grammatical and semantic information than the English words, the subject can be omitted in many cases, and verbs forms carry implicit conjuga-

tion, without additional words. That, coupled with the high number of meanings that the same word can have, increases the computational complexity for syntactic, semantic and morphological analyzers, which so behave differently in Spanish and English. Spanish is the third language in the world according to the number of speakers, after Mandarin and English, but in terms of studies related to NLP we have not found many scientific papers.

Impact of NLP and Semantics. Our experimental evaluation suggests that the influence of NLP and semantic tools is not quantitatively as important as the classic statistical approaches, although their contribution can tip the scales when evaluating the quality of a labeling technique, since the difference in terms of precision and recall is sufficiently influential (near 20%). So, our conclusion is that a statistical approach can be successfully complemented with semantic techniques to obtain an acceptable automatic categorisation. Our experience also proves that facing this issue in a real environment when professional results are needed, the typical machine learning approach is the best option but is not always enough. We have seen that it should be complemented with other techniques, in our case semantic and linguistic ones. Anyway, the main drawback of the semantic techniques is that the work of searching or constructing the ontologies for each set of tags of every topic, populating them, and building the relationship tables, is harder than the typical training of the machine learning approaches. So, although the results are better, the scalability could be problematic. Sometimes it can be quite costly, especially if detailed knowledge of the topic to tag is required in order to appropriately configure the system.

NLP Future Tasks In some categorisation scenarios, like bigger analysis (novels, reports, etc.) or groups of documents of the same field, it can be interesting to obtain a summary of the given inputs in order to categorise them with their general terms before entering a more detailed analysis which requires the entire texts. These summaries, alongside with the previous defined tasks, can lead to a more suitable detailed labelling, providing hints of which knowledge bases might be interesting to work with. In order to achieve this, we can perform syntactic analysis to simplify the sentences of the summaries, as we have seen in works like (Silveira and Branco, 2012), and then we will use the obtained results to filter unnecessary information and select the most relevant sentences without compromising the text integrity. Although the required structures have been implemented and some approaches as (Garrido et al., 2013a) are being designed and tested, they are into an early stage and they require more work before trying to use it inside the categorisation pipeline.

Open Tasks. As future work, we plan to increase the number of methods used in the pipeline, and to test this methodology in new contexts and languages. It is noteworthy that a piece of news is a very specific type of text, characterized by objectivity, clarity, and the use of synonyms and acronyms, the high presence of specific and descriptive adjectives, the tendency to use impersonal or passive constructions, and the use of connectors. Therefore it is not sufficient to test only with this kind of text, and to make a more complete study is necessary to work with other types. In fact, some tests have been made with GENIE with other types of documents very different from news, such as book reviews, business reports, lyrics, blogs, etc. and the results are very promising, but it is early to assert the generality of the solution in different contexts because the studies are still in progress.

ACKNOWLEDGEMENTS

This research work has been supported by the CICYT project TIN2010-21387-C02-02 and DGA-FSE. Thank you to Heraldo Group and Diario de Navarra.

REFERENCES

- Aguado de Cea, G., Puch, J., and Ramos, J. (2008). Tagging Spanish texts: The problem of 'se'. In *Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2321–2324.
- Amitay, E., Har'El, N., Sivan, R., and Soffer, A. (2004). Web-a-where: geotagging web content. In *27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 273–280. ACM.
- Atkinson, M. and Van der Goot, E. (2009). Near real time information mining in multilingual news. In *Proceedings of the 18th International Conference on World Wide Web, WWW '09*, pages 1153–1154. ACM.
- Bikakis, N., Giannopoulos, G., Dalamagas, T., and Sellis, T. (2010). Integrating keywords and semantics on document annotation and search. In *On the Move to Meaningful Internet Systems (OTM 2010)*, pages 921–938. Springer.
- Bloehdorn, S. and Hotho, A. (2006). Boosting for text classification with semantic features. In *Advances in Web mining and Web Usage Analysis*, pages 149–166. Springer.
- Bruno, M., Canfora, G., Di Penta, M., and Scognamiglio, R. (2005). An approach to support web service classification and annotation. In *2005 IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE'05)*, pages 138–143. IEEE.
- Carrasco, R. and Gelbukh, A. (2003). Evaluation of TnT Tagger for Spanish. In *Proceedings of ENC, Fourth Mexican International Conference on Computer Science*, pages 18–25. IEEE.
- Carreras, X., Chao, I., Padró, L., and Padró, M. (2004). FreeLing: An open-source suite of language analyzers. In *Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pages 239–242. European Language Resources Association.
- Chau, R. and Yeh, C.-H. (2004). Filtering multilingual web content using fuzzy logic and self-organizing maps. *Neural Computing and Applications*, 13(2):140–148.
- Elberrichi, Z., Rahmoun, A., and Bentaallah, M. A. (2008). Using WordNet for text categorization. *The International Arab Journal of Information Technology (IAJIT)*, 5(1):16–24.
- Garrido, A. L., Buey, M. G., Escudero, S., Ilarri, S., Mena, E., and Silveira, S. B. (2013a). TM-gen: A topic map generator from text documents. In *25th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2013), Washington DC (USA)*, pages 735–740. IEEE Computer Society.
- Garrido, A. L., Buey, M. G., Ilarri, S., and Mena, E. (2013b). GEO-NASS: A semantic tagging experience

- from geographical data on the media. In *17th East-European Conference on Advances in Databases and Information Systems (ADBIS 2013)*, Genoa (Italy), volume 8133, pages 56–69. Springer.
- Garrido, A. L., Gomez, O., Ilarri, S., and Mena, E. (2011). NASS: News Annotation Semantic System. In *23rd IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2011)*, Boca Raton, Florida (USA), pages 904–905. IEEE Computer Society.
- Garrido, A. L., Gomez, O., Ilarri, S., and Mena, E. (2012). An experience developing a semantic annotation system in a media group. In *Proceedings of the 17th International Conference on Applications of Natural Language Processing and Information Systems*, pages 333–338. Springer.
- Gilchrist, A. (2003). Thesauri, taxonomies and ontologies - an etymological note. *Journal of Documentation*, 59(1):7–18.
- Goodchild, M. F. and Hill, L. (2008). Introduction to digital gazetteer research. *International Journal of Geographical Information Science*, 22(10):1039–1044.
- Gruber, T. R. et al. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220.
- Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% solution. In *Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60. Association for Computational Linguistics.
- Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Tenth European Conference on Machine Learning (ECML'98)*, pages 137–142. Springer.
- Joachims, T. (2004). *SVM Light Version: 6.01*. <http://svmlight.joachims.org/>.
- Lee, S. O. K. and Chun, A. H. W. (2007). Automatic tag recommendation for the web 2.0 blogosphere using collaborative tagging and hybrid and semantic structures. *Sixth Conference on WSEAS International Conference on Applied Computer Science (ACOS'07)*, World Scientific and Engineering Academy and Society (WSEAS), 7:88–93.
- Leopold, E. and Kindermann, J. (2002). Text categorization with support vector machines. How to represent texts in input space? *Machine Learning*, 46:423–444.
- Li, H., Srihari, R. K., Niu, C., and Li, W. (2002). Location normalization for information extraction. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*, volume 1. Cambridge University Press.
- Maynard, D., Peters, W., and Li, Y. (2006). Metrics for evaluation of ontology-based information extraction. In *Workshop on Evaluation of Ontologies for the Web (EON) at the International World Wide Web Conference (WWW'06)*.
- McGuinness, D. L., Van Harmelen, F., et al. (2004). OWL web ontology language overview. *W3C recommendation 10 February 2004*.
- Miller, G. A. (1995). WordNet: a lexical database for english. *Communications of ACM*, 38(11):39–41.
- Mishra, R. B. and Kumar, S. (2011). Semantic web reasoners and languages. *Artificial Intelligence Review*, 35(4):339–368.
- Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):10:1–10:69.
- Quercini, G., Samet, H., Sankaranarayanan, J., and Lieberman, M. D. (2010). Determining the spatial reader scopes of news sources using local lexicons. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 43–52. ACM.
- Rauch, E., Bukatin, M., and Baker, K. (2003). A confidence-based framework for disambiguating geographic terms. In *HLT-NAACL 2003 Workshop on Analysis of Geographic References, vol. 1*, pages 50–54. Association for Computational Linguistics.
- Resnik, P. (1999). Disambiguating noun groupings with respect to WordNet senses. In *Natural Language Processing Using Very Large Corpora*, pages 77–98. Springer.
- Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.
- Scharkow, M. (2013). Thematic content analysis using supervised machine learning: An empirical evaluation using German online news. *Quality and Quantity*, 47(2):761–773.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.
- Sekine, S. and Ranchhod, E. (2009). *Named Entities: Recognition, Classification and Use*. John Benjamins.
- Shen, D., Sun, J.-T., Yang, Q., and Chen, Z. (2006). A comparison of implicit and explicit links for web page classification. In *Proceedings of the 15th International Conference on World Wide Web, WWW '06*, pages 643–650. ACM.
- Silveira, S. B. and Branco, A. (2012). Extracting multi-document summaries with a double clustering approach. In *Natural Language Processing and Information Systems*, pages 70–81. Springer.
- Siolas, G. and d'Alché Buc, F. (2000). Support vector machines based on a semantic kernel for text categorization. In *IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN 2000)*, volume 5, pages 205–209. IEEE.
- Smeaton, A. F. (1999). *Using NLP or NLP Resources for Information Retrieval Tasks*. *Natural Language Information Retrieval*. Kluwer Academic Publishers.
- Trillo, R., Gracia, J., Espinoza, M., and Mena, E. (2007). Discovering the semantics of user keywords. *Journal of Universal Computer Science*, 13(12):1908–1935.
- Vossen, P. (1998). *EuroWordNet: A multilingual database with lexical semantic networks*. Kluwer Academic Boston.
- Wilbur, W. J. and Sirotkin, K. (1992). The automatic identification of stop words. *Journal of Information Science*, 18(1):45–55.

Comparing Topic Models for a Movie Recommendation System

Sonia Bergamaschi, Laura Po and Serena Sorrentino

DIEF, University of Modena and Reggio Emilia, 41125 Modena, Italy
{name.surname}@unimore.it

Keywords: Movie Recommendation System, LDA, LSA.

Abstract: Recommendation systems have become successful at suggesting content that are likely to be of interest to the user, however their performance greatly suffers when little information about the users preferences are given. In this paper we propose an automated movie recommendation system based on the similarity of movie: given a target movie selected by the user, the goal of the system is to provide a list of those movies that are most similar to the target one, without knowing any user preferences. The Topic Models of Latent Semantic Allocation (LSA) and Latent Dirichlet Allocation (LDA) have been applied and extensively compared on a movie database of two hundred thousand plots. Experiments are an important part of the paper; we examined the topic models behaviour based on standard metrics and on user evaluations, we have conducted performance assessments with 30 users to compare our approach with a commercial system. The outcome was that the performance of LSA was superior to that of LDA in supporting the selection of similar plots. Even if our system does not outperform commercial systems, it does not rely on human effort, thus it can be ported to any domain where natural language descriptions exist. Since it is independent from the number of user ratings, it is able to suggest famous movies as well as old or unheard movies that are still strongly related to the content of the video the user has watched.

1 INTRODUCTION

Recommendation systems are information filtering systems that recommend products available in e-shops, entertainment items (books music, videos, Video on Demand, books, news, images, events etc.) or people (e.g. on dating sites) that are likely to be of interest to the user. These system are the basis of the targeted advertisements that account for most commercial sites revenues. In the recent years, some events catalized the attention on movie recommendation systems: in 2009, a million-dollar prize has been offered by the DVD rental site Netflix¹ to anyone who could improve their predictions by 10%²; in 2010 and 2011 we saw the International Challenges on Context-Aware Movie Recommendation³; moreover, in 2013, Netflix announced a new developer contest called the “Netflix Cloud Prize”⁴ which is promising a prize money of \$100,000 to those who improve the performance of Netflix’s cloud computing services.

¹<http://www.netflixprize.com/>

²The grand prize was given to the BellKor’s Pragmatic Chaos team which bested Netflix’s own algorithm for predicting ratings by 10.06%.

³<http://2011.camrachallenge.com/>

⁴<https://github.com/Netflix/Cloud-Prize/wiki>

Recommendation systems have become relatively successful at suggesting content, however their performance suffers greatly when little information about the user’s preferences is given. These situations are not rare; they usually occur when the users are new to a system, the first time a system is launched on the market (no previous users have been logged), for new items (where we do not have any history on preferences yet) (Adomavicius and Tuzhilin, 2005) and when, because of user desires for privacy, the system does not record their preferences (Rashid et al., 2008). In such cases, making suggestions entirely based on the content that is being recommended can be a good solution.

The focus of the paper is to provide an automatic movie recommendation system that does not need any a priori information about users. The paper compares two specific techniques (LDA and LSA) that have been implemented in our content-based recommendation system. Although topic models are not new in the area of recommendation systems, their use has not been deeper analyzed in a specific domain, such as the movie domain. Our intention is to show how these well-known techniques can be applied in such specific domain and how they perform.

The automatic movie recommendation system

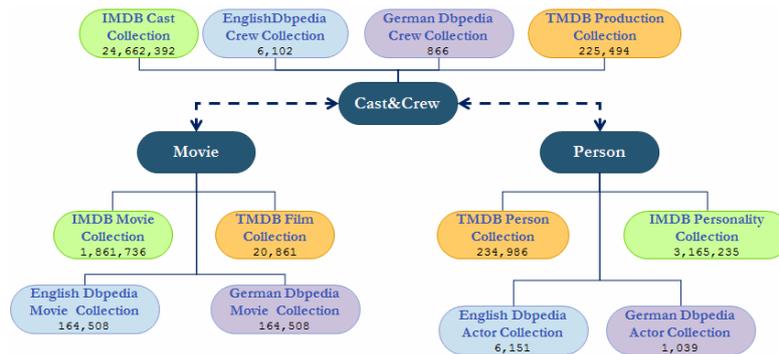


Figure 1: The local MongoDB database.

permits, given a movie, to supply users with a list of those movies that are most similar to the target one. The way the system detects the list of similar movies is based upon an evaluation of similarity among the plot of the target movie and a large amount of plots that is stored in a movie database. The movie database has been constructed by integrating different movie databases in a local NoSQL (MongoDB) database, building a collection of about two hundred thousand plots together with the most relevant movie metadata.

The context where our system works is that of video-on-demand (VOD). Generally speaking, this is the case when a user is looking for an item without being registered on the site in which he is looking for (searching a book on Amazon, a movie on IMDb etc.). We assumed the only information we have about the user is his first choice, the movie he has selected/he is watching (we do not have a history about his past selections nor a profile about his general interests). When watching a VOD movie, users explicitly request to buy and to pay for that movie, then what our system attempt to do is proposing a list of similar movies assuming that the chosen film has been appreciated by the user (the system assumes the user liked the movie if his play time is more then 3/4 of the movie play time). Here, we also assume that we have no knowledge about the preferences of the users; namely, about who is watching the film, and also with regard to other users who have previously accessed the system.

There are dozens of movie recommendation engines on the web. Some require little or no input before they give you movie titles, while others want to find out exactly what your interests are, however all of these systems rely on ratings directly or indirectly expressed by users of the system (some examples are *Netflix*, *Rotten Tomatoes*, *Movielens*, *IMDb*, *Jinni*).

Starting from our previous work (Farinella et al., 2012) that highlighted the power of using the Latent Semantic Analysis (LSA) in contrast with weighting

techniques (such as log and tf-idf) in suggesting similar movies, in this paper we intend to integrate in the system the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and then, by evaluating on real users we compare the performance of our system based on LSA and LDA.

In (Griffiths et al., 2007) it has been shown that the Latent Dirichlet Allocation (LDA) Topic Model (Blei et al., 2003) outperforms LSA, in the representation of ambiguous words and in a variety of other linguistic processing and memory tasks. Hereby, we intend to analyze the performance of LDA on the movie domain and compare its behaviour with regard to LDA. In the end, we examine the performance of our system with regard to a commercial approach.

The system has been developed in collaboration between the database group of the University of Modena and Reggio Emilia⁵ and vfree.tv⁶, a young and innovative German company focused on creating new ways of distributing television content and generating an unprecedented watching experience for the user.

The paper is structured as follows. Section 2 describes the structure of the local movie database MongoDB. Section 3 describes hoe the system performs the similarity computations among movie plots by using the LDA and LSA Topic Models. The experimental results of this study are presented in Section 4: we show the computational costs of building the LSA and LDA matrices, and the results of off-line tests performed on three recommendation systems (LSA, LDA and a commercial approach). Section 5 presents some related work. Conclusion and future work are depicted in Section 6.

⁵<http://www.dbgroup.unimo.it>

⁶<http://vfree.tv>

| | | |
|--|--|---|
| <pre> IMDB Personality Collection { "_id": "4d947a25d69948679409f92d", "person_id": "Spielberg, Steven", "fullname": "Steven Spielberg" } English DBPEDIA Actor Collection { "_id": "4e43dd0a7a79ae38060007c7", "person_id": "http://dbpedia.org/resource/Steven_Spielberg", "fullname": "Steven Spielberg" } </pre> | <pre> IMDB Cast Collection { "_id": "4d99c384d699481a2925292a", "person_id": "Spielberg, Steven", "work_as": ["info": ["attributes": "(producer)", "character": ""]], "role": "producer" } English DBPEDIA Crew Collection { "_id": "4e54fca7a79ae3cbf0007ab", "movie_id": "http://dbpedia.org/resource/Schindler%27s_List", "person_id": "http://dbpedia.org/resource/Steven_Spielberg", "work_as": ["info": ["role": "director"], "info": ["role": "producer"]] } </pre> | <pre> IMDB Movie Collection { "_id": "4d942fb4d6994808470593e2", "aka_title": ["Schindler's List - La lista di Schindler", "Schindlers Liste", "Steven Spielberg's Schindler's List"], "genre": ["Biography", "Drama", "History", "War"], "movie_id": "Schindler's List (1993)", "plot": ["Oskar Schindler is a vainglorious and greedy German businessman who becomes unlikely humanitarian amid the barbaric Nazi reign when he feels compelled to turn his factory [...]"], "rating": 8.9, "release_year": "1993", "short_title": "Schindler's List", "title": "Schindler's List (1993)" } English DBPEDIA Movie Collection { "_id": "4e43a5017a79ae07520004bc", "aka_title": "Schindlers Liste", "budget": "2.2E7", "distributor": "Universal Studios", "gross": "3.21E8", "movie_id": "http://dbpedia.org/resource/Schindler%27s_List", "plot": "Schindler's List is a 1993 American epic drama film about Oskar Schindler, a German businessman who saved the lives of more than a thousand Polish-Jewish refugees [...]", "title": "Schindler's List", "type": "movie" } German DBPEDIA Movie Collection { "_id": "4e43a5017a79ae07520004bd", "aka_title": "Schindler\\u2019s List", "lang": "Hebräische Sprache", "location": "USA", "movie_id": "http://dbpedia.org/resource/Schindler%27s_List", "release_year": "1993", "plot_de": "Schindlers Liste ist ein Spielfilm von Steven Spielberg aus dem Jahr 1993 nach dem gleichnamigen Roman (im Original Schindler's Ark) von Thomas Keneally. Thomas Keneally beschreibt in dem Buch Schindlers Liste, [...]", "title": "Schindler's List", "type": "movie" } </pre> |
|--|--|---|

Figure 2: Documents related to the “Schindler’s List” movie.

2 THE MOVIE DATABASE

The principal aim of a local repository of movies is to supply an extensive and reliable representation of multimedia that can be queried in a reasonable time. The local database of our system has been defined, as in our previous work (Farinella et al., 2012), by importing data from external repositories. In particular, we selected the Internet Movie Database (IMDb)⁷, DBpedia⁸ and the Open Movie Database (TMDb)⁹. Since local database needs to easily import data from different sources and perform queries on a huge amount of data (thousands of movies) in a short time, we chose MongoDB¹⁰, a non relational database and schema-free. MongoDB features allow to create databases with flexible and simple structure without decreasing the time performance when they are queried.

Information about movies can be classified in either information that are related to multimedia or information that are about people that participated in the production of multimedia. This led to the creation of three main databases, each storing collections from the 4 sources (as shown in Figure 1). As MongoDB do not enforce document structure, this flexibility allows an easy adaptation to integrate different/new datasets

⁷<http://www.imdb.com/>

⁸<http://dbpedia.org/>

⁹<http://www.themoviedb.org/>

¹⁰<http://www.mongodb.org/>

into the system. A single collection can store documents with different fields. Thus there cannot really be a description of a collection, like the description of a table in the relational databases. However, to give an insight into the local DB, we extracted some documents that store information related to the “Schindler’s List” movie. In figure 2, documents from different collections (IMDb, and the English and German version of DBpedia) are shown. It can be noticed how the information are heterogeneously represented in the collections (see the different information stored in the English and German version of DBpedia Movie collection) and how flexible the structure of each documents is (see for example in the English DBpedia Crew Collection the double role of Steven Spielberg that is both the director and the producer of the movie).

3 PLOT SIMILARITY COMPUTATION

The similarity of two media items depends on their features likeness. Hence, for each feature, a specific metric is defined in order to compute a similarity score. Most of the metrics that are adopted are calculated through only few simple operations. However, if we want to consider also movie plots, the similarity computation becomes more complex. Our approach is based on the Vector Space Model (VSM) (Salton

et al., 1975), this model creates a space in which both documents and queries are represented by vectors. In the area of the Information Retrieval the VSM has been considered as one of the most effective approaches, and its behaviour has also been studied applied on recommendation systems (Musto, 2010).

Our system takes advantage of this model to represent the different movie plots: each plot (or document from now on) is represented as a vector of keywords with associated weights. These weights depend on the distribution of the keywords in the given training set of plots that are stored in the database. Vectors representing plots are then joined in a matrix representation where each row corresponds to a plot and each column corresponds to a keyword extracted from the training set plots (i.e. the *document by keyword matrix*). Thus, each cell of the matrix represents the weight of a specific keyword according to a specific plot.

The matrix computation goes through four main steps:

1. *Plot Vectorization* - relevant keywords are extracted and then stop words removal and lemmatization techniques are applied;
2. *Weights Computation*- weights are defined as the occurrences of keywords in the plots; the initial weights are then modified by using the tf-idf technique (Salton et al., 1975)(but other suitable weighting techniques could be used as well), thus building the *document by keyword matrix*;
3. *Matrix Reduction by using Topic Models* - the *document by keyword matrix* is reduced to a lower dimensional space by using the Topic Models LDA and LSA, thus it is transformed into a *document by topic matrix*.
4. *Movie Similarity Computation*- starting from the *document by topic matrix*, the similarity between two plots is computed by considering their topics as features instead of words.

3.1 Plot Vectorization

If two plots are to be compared, they will need to be converted into vectors of keywords. As preliminary operations, keyword extraction and filtering activity are performed. Keywords correspond to terms within the document that are representative of the document itself and that, at the same time, are discriminating. Less discriminative words, the so called *stop words*, are discarded, while the other terms are preprocessed and substituted in the vector by their lemmas (*lemmatization*).

Lemmatization and keyword extraction are performed by using *TreeTagger*¹¹, developed at the Institute for Computational Linguistics of the University of Stuttgart. This tool can annotate documents with part-of-speech and lemma information in both English and German language.

Keywords extracted from plots as well as their local frequencies (occurrences in the description of the plot) are stored as features of the media item in the local database MongoDB. This choice has been made for two main reasons. First, the keyword extraction process is relatively slow¹² compared to the access to database values. Since the weighting techniques are based on the global distribution of the keywords over the whole corpus of plots, it is necessary to generate all the vectors before applying the weighting technique. Second, while weights change when new multimedia plots are added into the system, the local keyword occurrences do not.

3.2 Weights Computation

Weighting techniques are used for computing keyword weights. A weight is a value in the range $[0, 1]$ that represents the relevance of a specific keyword according to a specific document. A weight is calculated on the basis of the local distribution of the keyword within the document as well as on the global distribution of the keyword in the whole corpus of plots. Keywords with a document frequency equal to 1 are discarded. Since, our previous work (Farinella et al., 2012) has compared *tf-idf* and *log* weighting techniques revealing that the results are very similar, in this paper we employ only the tf-idf technique for computing the weights.

3.3 Matrix Reduction by using Topic Model

The Vector Space Model treats keywords as independent entities. To find documents on specific concepts, we must provide the correct key terms. This representation leads to several issues: (1) there can be an high number of keywords when we have to deal with a huge amount of documents; (2) if any keyword changed, the document would not convey the same concept.

These problems can be faced by a representation into a Topic Model (Park and Ramamohanarao, 2009).

¹¹<http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>

¹²One database access, using MongoDB, takes about 0.3 milliseconds while the extraction of keywords from a plot takes more than one second.

The Topic Model explores the idea that the concept held by a set of terms can be represented as a weighted distribution over a set of topics. Each topic is a linear combination of terms, where to each term a weight reflecting the relevance of the term for that topic is associated. For example, high weights for *family* and *house* would suggest that a topic refers to a social unit living together, whereas high weights for *users* and *communication* would suggest that a topic refers to social networking.

Topics can be found by clustering the set of keywords. The use of topics drastically reduces the dimension of the keyword Matrix obtained by Vector Space Model. Moreover, if a keyword changes, the document conveys the same idea as long as the new keyword is taken from the same topic pool.

Topic vectors may be useful in the context of movie recommendation systems for three main reasons: (1) the number of topics that is equal to the number of non-zero eigenvectors is usually significantly lower than the number of keywords, the topic representation of the plots is more compact¹³; (2) the topic representation of the keywords makes possible to add movies that have been released after the definition of the matrix without recomputing it;

(3) to find similar movies starting from a given one, we just need to compute the topic vectors for the plot of the movie and then compare these vectors with the ones we have stored in the matrix finding the top relevant.

The main Topic Models exploited so far in literature are the LSA (also called Latent Semantic Indexing (LSI)) and the LDA. In the following, we briefly describe both the methods and how they can be applied to our movie recommendation system.

3.3.1 Latent Semantic Analysis (LSA)

LSA is a model for extracting and representing the contextual-usage meaning of words by statistical computations applied to a large corpus of documents.

The LSA consists of a *Singular Value Decomposition* (SVD) of the matrix T (Training set matrix) followed by a *Rank lowering* (for more details see (Dumais, 2004; Deerwester et al., 1990)). The Singular Value Decomposition consists of representing the matrix T , the *document by keyword matrix* that represents the relationships between keywords and plots, as the product of three matrices: K, S, D^T . Matrix K , the *topic by keyword matrix*, represents the relationships between keywords and topics, while matrix S is a diagonal matrix whose values represent the square

¹³Thus, we store the matrix of document-topic vectors to represent the training set.

roots of the so called eigenvalues of the matrix TT^T . Matrix D^T , *document by topic matrix*, represents the relationships between plots and topics.

The Singular Value Decomposition is consequently followed by a *Rank lowering* by which the matrices S and T are transformed respectively into the matrices S', T' . The purpose of dimensionality reduction is to reduce *noise* in the latent space, resulting in a richer word relationship structure that reveals latent semantics present in the collection. In a LSA system, the matrices are truncated to z dimensions (i.e. topics). The optimal z is determined empirically for each collection. In general, smaller z values are preferred when using LSA, due to the computational cost associated with the SVD algorithm, as well as the cost of storing and comparing large dimension vectors¹⁴.

3.3.2 Latent Dirichlet Allocation (LDA)

LSA provides a simple and efficient procedure for extracting a topic representation of the associations between terms from a term-document co-occurrence matrix. However, as shown in (Griffiths et al., 2007), this representation makes it difficult for LSA to deal with the polysemous terms. The key issue is that its representation does not explicitly identify the different senses of a term. To address this problem we investigated the use of the Latent Dirichlet Allocation (LDA) Topic Model.

Unlike LSA, LDA is a probabilistic Topic Model, where the goal is to decompose a conditional *term by document probability distribution* into two different distributions, this allows each semantic topic z to be represented as a multinomial distribution of terms, and each document d to be represented as a multinomial distribution of semantic topics. The model introduces a conditional independence assumption that document d and keyword k are independent conditioned on the hidden variable, topic z .

LDA can also be interpreted as matrix factorization where document over keyword probability distribution can be split into two different distributions: the topic over keyword distribution, and the document over topic distribution. Thus, it appears clear, that we can easily make a direct correspondence between the document by topic matrix obtained from LSA and the document over topic distribution obtained by using LDA.

Both LDA and LSA permit to find a low dimensional representation for a set of documents with re-

¹⁴In our previous work we determined 500 as a good number of topic. This value allows have reasonable computational costs, and maintains an appropriate level of accuracy.

gard to the simple term by document matrix. This dimensionality in both cases has to be decided a priori. By adopting LSA, we were able to represent each plot of the IMDb database with 500 topics, instead of 220,000 keywords (Farinella et al., 2012). For LDA (which has been demonstrated working well for a number of topics over 50 (Blei et al., 2003)), after a few experimental evaluations, we decided to use 50 topics.

3.4 Movie Similarity Computation

As previously described by using LSA or LDA the *document by keyword matrix* is decomposed into several matrices. The *document by topic matrix* is the one that is used to represent the movie of our database in a lower dimensional space and also to compute the similarity score between two plots.

To calculate the similarity score between two documents we use the cosine similarity. This metric is used to either compare plots within the training set or plots that are not included in the training set.

Definition - Cosine Similarity: *Given two vectors v_i , and v_j , that represent two different plots, the cosine angle between them can be calculated as follows:*

$$\cosin(v_i, v_j) = \frac{\sum_k (v_i[k] \cdot v_j[k])}{\sqrt{\sum_k v_i[k]^2} \cdot \sqrt{\sum_k v_j[k]^2}}$$

The value of the cosine angle is a real number in the range $[-1, 1]$. If the cosine is equal to 1 the two vectors are equivalent, whereas if it is -1 the two vectors are opposite.

The similarity of plots can also be combined with the similarity of other features such as directors, genre, producers, release year, cast etc.

Definition - Feature-based Similarity: *Given two media items (m_1 and m_2) the feature-based similarity is defined as a weighted linear combination of the similarity of the feature values that describe the two items:*

$$\text{sim}(m_1, m_2) = \sum_{i=1}^{FN} w_i \cdot \text{sim}_i(f_{1,i}, f_{2,i})$$

where FN is the number of features that describe a media item, sim_i is the metric used to compare the i -th feature, $f_{j,k}$ is the value assumed by feature k in the j -th media item. The result of each metric is normalized to obtain a value in the range $[0, 1]$ where 1 denotes equivalence of the values that have been compared and 0 means maximum dissimilarity of the values (Debnath et al., 2008).

4 EXPERIMENTS

We have performed several tests in order to evaluate our system, the goal was to compare the effectiveness of LDA and LSA techniques and to evaluate the performance of the system on real users.

Our previous research (Farinella et al., 2012) has shown that:

- There is not a big difference in the results obtained by applying log or tf-idf weighting techniques. Thus, we can use one of them.
- The use of the Topic Model LSA shows a noticeable quality improvement compared to the use of the SVD model. LSA allows to select plots that are better related to the target's plot themes.

Starting from these results, we conducted new tests and evaluations of the system. First of all, we loaded data from IMDb into the local database MongoDB and evaluated the computational costs of building the LSA and LDA matrices. Then, we compared the two Topic Models manually, by analyzing their behaviours in some special cases. Finally, we conducted off-line tests. We built two surveys asking real users to judge the similarity of each film in a list with regard to a target movie. The first test compared the performance of LDA and LSA. The second test compared the performance of LSA and a commercial system, IMDb. A third test evaluates the precision of the three recommendation systems.

4.1 Setup of the Environment

The DataBase Management System used is MongoDB 2.4.1, the system has been installed on a machine with the following characteristics: OS: Windows Server 2008 R2 64-bit; CPU: Intel (R) Xeon E5620 Ghz 2:40; RAM: 12 GB. The 64bit version of MongoDB is necessary as it is the only version that allow working with databases greater than 2 GB.

A virtual machine for the execution of the code has been installed on the server. The virtual machine has the following features: OS: Ubuntu 12.04 LTS 64-bit; RAM: 8 GB; 20 GB dedicated to the virtual hard disk; 4 cores. The virtual machine has been set up with VMWare Workstation 9.0.1¹⁵. The 32-bit architecture makes it possible to instantiate a maximum of 2 GB of memory for a process. The creation of the LSA and LDA models exceeds the threshold of 2 GB, then the use of a 64-bit architecture is crucial in order to avoid memory errors.

¹⁵<http://www.vmware.com/products/workstation/>

4.2 Evaluation of the Computational Costs

The SVM of the plot-keyword matrix have a complexity of $O(d \times k)$ where d is the number of multimedia (rows of the matrix) and k is the number of keywords and $d \geq k$. There are about 1,861,736 multimedia in the IMDb database, but only for 200,000 there is a plot available. These plots contain almost 220,000 different keywords. Thus, the time cost for the decomposition of the matrix is $O = 3 \cdot 10^{15}$. Furthermore, the decomposition requires random access to the matrix, which implies an intensive usage of the central memory.

Both LSA and LDA decrease this cost by using a reduced matrix. The Document by Topic Matrix used by LSA has a dimension of $d \times z$ where z is the number of topic (columns). The Document Distribution over Topic Matrix used by LDA has a dimension of $d \times z$. Usually LSA requires more topics than LDA. Thus, the cost for the computation of the LDA matrix is further decreased. In order to avoid the central memory saturation, we employ the framework Gensim¹⁶. It offers functions to process plain document including algorithms performing both LSA and LDA which are independent from the training corpus size.

Table 1 shows the computational costs to create the LSA and LDA models (the cost refers to the environment that we described in 4.1).

Table 1: Computational Costs.

| Operation | Time (minutes) | CPU avg use | Memory avg use |
|----------------|----------------|-------------|----------------|
| Plot vect. | 5 | 75% | 11% |
| Tf-idf weights | 1 | 97% | 10% |
| LSA weights | 120 | 97% | 42% |
| LDA weights | 60 | 95% | 40% |

Table 2: LSA and LDA Topic Model comparison.

| Configuration | LSA | LDA |
|----------------------|---------------|--------------|
| min. document freq. | 10 | 10 |
| min. vector length | 20 | 20 |
| min. tf-idf weight | 0.09 | 0.09 |
| min. lsa/lda weight | 0.001 | 0.001 |
| n. of topics | 500 | 50 |
| matrix size | 204285 x 500 | 204285 x 50 |
| Similarity time cost | 12 sec | 6 sec |

Table 2 summarizes the configuration adopted for LSA and LDA and the time performance of the topic models when, starting from a given plot, they rank all the other plots in the database. Since the LDA model requires less topics (50 instead of the 500 required by

¹⁶<http://radimrehurek.com/gensim/>

LSA), it has a computation cost and a similarity time cost lower than the ones for LSA.

Table 3: A comparison between LSA and LDA techniques on the movie "Batman Begins".

| LSA | LDA |
|----------------------------------|----------------------------|
| 1.Batman Begins(2005) | 1.Batman Begins(2005) |
| 2.Batman(1989) | 2.Batman Forever(1995) |
| 3.Batman:Gotham Knight(2008) | 3.The Dark Knight(2008) |
| 4.The Batman/Superman Hour(1968) | 4.The Batman(2004) |
| 5.The Batman(2004) | 5.Frankie and Johnny(1991) |
| 6.The Dark Knight(2008) | 6.The Exorcist(1973) |

Table 4: A comparison between LSA and LDA techniques on the movie "The Matrix".

| LSA | LDA |
|--------------------------------|---------------------------------|
| 1.The Matrix (1999) | 1.The Matrix (1999) |
| 2.Computer Warriors (1990) | 2.The Matrix Reloaded (2003) |
| 3.Electric Dreams (1984) | 3.Simulacrum (2009) |
| 4.Willkommen in Babylon (1995) | 4.Virus X (2010) |
| 5.TRON 2.0 (2003) | 5.Fallen Moon (2011) |
| 6.Hackers (1995) | 6.The Matrix Revolutions (2003) |

4.3 Topic Model Comparison

We have performed several tests in order to evaluate which of the Topic Models was the best in defining the recommended movie list.

As described in section 3, the similarity of plots can be combined with the similarity of other features. We decided not to consider the features for this evaluation, so the LSA and LDA are compared considering only plots and none of the movie features. This decision was gained after a manual evaluation of the results of LDA and LSA with or without features. The manual evaluation showed several problems that have led us to formulate some considerations: (1) for each movie we have a broad list of actors, as IMDb reports the complete list including the background actors. Evaluating similarity starting from this list is really complicated and might lead to meaningless results; (2) in order to decrease computation cost, the features have been applied as a filter after the computation of the top 100 most similar movies according to the plot. Thus, the application of the features do not always improve the results gained by LSA or LDA.

To examine the quality of results of LDA and LSA, we chose three movies: two movies of a saga and a movie without a sequel. We calculated the five most similar movies for each target movie and analyzed the outcome. Table 3 shows the results for

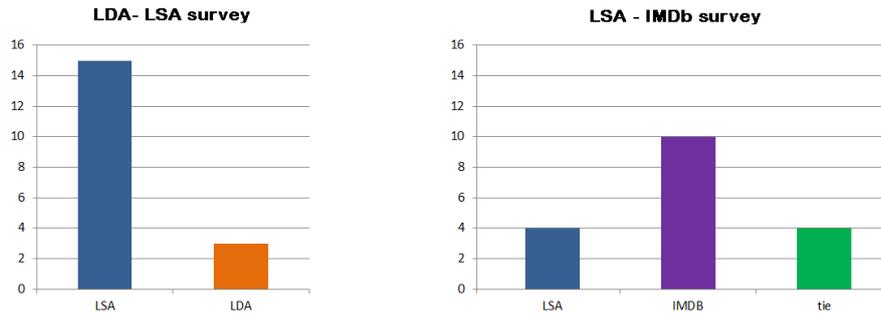


Figure 3: Performance of the topic models and IMDb on the two surveys.

the target movie “Batman Begins”. All movies recommended by LSA belong to the series of Batman. The LDA list contains many movies of the saga, however the movies in fifth and sixth position have nothing in common with the target movie, these movies have a poor and short plot with words that are present also in the Batman Begins’s plot (such as evil, sinister forces, family, prison). For the second movie “The Matrix” (see Table 4), LSA selected movies referring to the topics of computer, network, programmer, hacker. The outcome of the LDA technique showed two movies of the trilogy and other movies containing terms and names that appear also in the target plot, but that do not refer to similar topics. The quality of the outcome decreases with movies that do not have a sequel as it can be seen in Table 5. For this kind of movies is difficult to evaluate the recommended movie list. For this reason we built a survey of popular movies that do not have a sequel and asked to real users to judge the similarity of the recommended movies.

Table 5: A comparison between LSA and LDA techniques on the movie “Braveheart”.

| LSA | LDA |
|--|-----------------------------------|
| 1.Braveheart (1995) | 1.Braveheart (1995) |
| 2.The Enemy Within (2010) | 2.Windwalker (1981) |
| 3.Journey of a Story (2011) | 3.Lipgloss Explosion(2001) |
| 4.Audition (2007) | 4.Race for Glory (1989) |
| 5.The Process (2011) | 5.Voyager from the Unknown (1982) |
| 6.Comedy Central Roast of William Shatner (2006) | 6.Elmo Saves Christmas (1996) |

4.4 Testing the Recommendation System with Real Users

In order to evaluate the performance of our recommendation system, we identified two crucial steps: first it is necessary to understand which of the two

Topic Models is more appropriate in the movie domain, then, we need to estimate its behaviour next to a commercial recommendation system, as IMDb.

We defined three off-line tests: the first collecting the recommendations of LDA and LSA for 18 popular movies (excluding sagas), the second comparing the recommendations of the best Topic Model with respect to the recommended movie list of IMDb, the third analyzing in more detail the preferences expressed by 5 users on the three recommendation systems. We asked users to fill out the survey by selecting the films that looked similar to the film in question. These evaluations have enabled us to draw some conclusions on the performance of the implemented Topic Models and on our system in general.

4.4.1 LDA versus LSA

The first off-line experiment involved 18 movies; for each of these movies, we selected the top 6 movies in the recommendation lists of both LSA and LDA. In order to propose results that can be easily judged by users, we discarded from the recommended movie lists: tv series, documentaries, short films, entries whose released year is before 1960, entries whose title is reported only in the original language, entries whose plot contains less than 200 characters.

We presented this list to users in a random order and asked them to judge for each movie in the list if it is similar to the target one, users can reply by choosing among “similar”, “not similar” and “I do not know” (see the survey displayed in left part of Figure 5). We collected 594 evaluations from 20 users in total. The evaluation results are shown in Table 6.

Table 6: A user evaluation of the two Topic Models.

| judgement | LSA | LDA |
|--------------------------|-----|-----|
| “similar” | 201 | 61 |
| “not similar” | 276 | 410 |
| “I do not know” | 120 | 129 |
| movies without judgement | 303 | 300 |

We also evaluated the behavior of the Topic Mod-

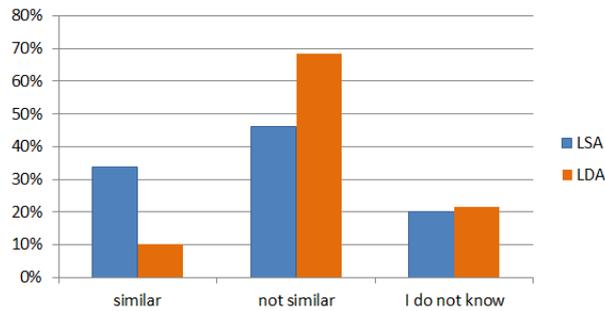


Figure 4: Percentage of users' judgements on LSA-LDA survey.

els on each film: on the 18 movies, we found that in 15 cases LSA selected the best recommendations and in 3 cases LDA selected the best recommendations (see left part of Figure 3). As expected from the previous comparison of the two models (reported in Tables 3,4,5), LSA supplied better recommendations than LDA. In Table 6 we have reported the total number of user judgements received (here we do not consider the movies for which users have not expressed a judgement).

4.4.2 LSA versus IMDb

In order to compare our system with respect to IMDb, we built another survey collecting recommendations for 18 popular movies (different with respect to the ones used in the LDA comparison): we selected them from the top 250 movies of IMDb¹⁷). Also in this case, we extracted only the top 6 movies in the recommendation lists of both LSA and IMDb.

In the previous survey, we obtained many void answers (i.e. on several recommended movies users do not expressed any opinion), moreover, some users highlighted that filling out the entire survey was very time consuming. Therefore, we decided to limit the options only to "similar".

We presented this list to users in a random order and asked users to judge for each movie in the list if it is similar to the target one (see the survey displayed in right part of Figure 5). The experiment has been conducted on 30 test participants. We collected 146 evaluations from 30 users in total. On the 18 movies, we found that in 4 cases LSA selected the best recommendations, in 10 cases IMDb selected the best recommendations and in 4 cases both systems showed the same performances (see right part of Figure 3).

¹⁷<http://www.imdb.com/chart/top>

4.4.3 User Preference Evaluation

We added an in-deep evaluation of the users preferences for the 18 popular movies used in 4.4.2. This evaluation has been based on the precision measure computes by using the classification of recommendation results introduced in (Gunawardana and Shani, 2009) (see table 7) as

$$Precision = \frac{\#tp}{\#tp + \#fp}$$

Table 7: Classification of the possible results of a recommendation of an item to a user (Gunawardana and Shani, 2009).

| | Recommended | Not recommended |
|---------------|------------------------------|------------------------------|
| Preferred | True-Positive (<i>tp</i>) | False-Negative (<i>fn</i>) |
| Not preferred | False-Positive (<i>fp</i>) | True-Negative (<i>m</i>) |

On the 18 movies, we examine punctual preferences expressed by 5 expert users on the top 6 items of the recommendation list, for this evaluation we consider the "similar" and "not similar" judgement expressed by the users. Thus for each recommendation list we calculate the precision of the system based on the user judgement.

We computed the average precision among users (AVG_P@6) and the standard deviation among the movies (DEV_M_P@6) and the users (DEV_U_P@6) (see table 8). AVG_P@6 reflects the average ratio of the number of relevant movies over the top-6 recommended movies for all users.

We found that the precision of LDA is quite low (about half as much as the LSA precision), while both LSA and IMDb reach a good precision. From this preliminary evaluation (that is quite limited since it is performed only on 5 users), it seems that the average precision on the entire set of movies of LSA is quite the same as the precision of IMDb. As it can be noticed, there is however a strong deviation of the precision value among different movies.

Table 8: Precision of the systems based on a punctual user preference evaluation.

| | AVG_P@6 | DEV_M_P@6 | DEV_U_P@6 |
|------|---------|-----------|-----------|
| LDA | 0.215 | 0.163 | 0.133 |
| LSA | 0.468 | 0.258 | 0.056 |
| IMDb | 0.416 | 0.281 | 0.064 |

4.5 Results and Discussion

Based on the results of the above-mentioned experiments, we can draw some conclusions:

- LDA does not have good performance on movie recommendations: it is not able to suggest movies

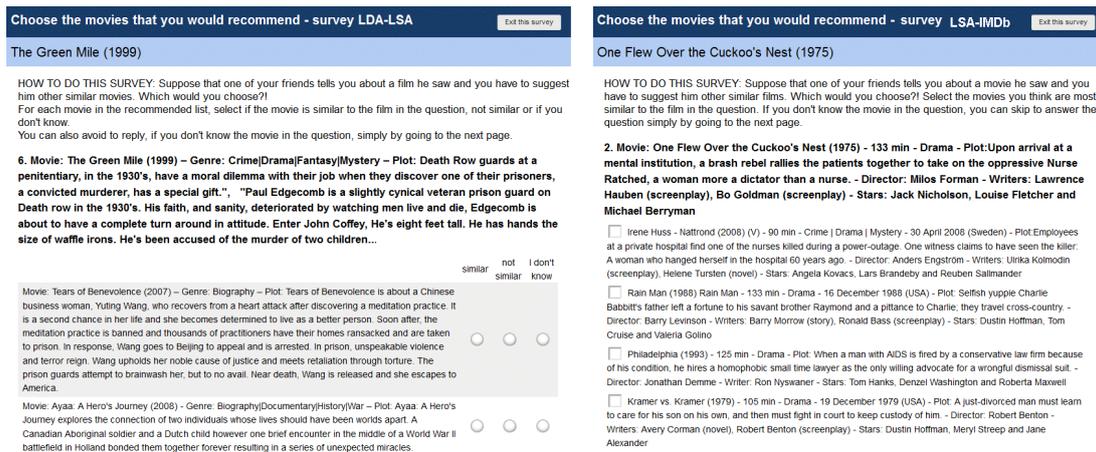


Figure 5: A screenshot of the beginning of a page of the surveys: LDA-LSA survey on the left, and LSA-IMDb survey on the right.

of the same saga and it suggests erroneous entries for movies that have short plot with words that are present also in the plot of the target movie (Sect.4.3), also the user evaluation underlines poor quality of the LDA recommendations (Sect.4.4.1,4.4.3);

- LSA achieves good performance on movie recommendations: it is able to suggest movies of the same saga and also unknown movies related to the target one (Sect.4.3), also the user evaluation underlines the good quality of the LSA recommendations (Sect.4.4.1,4.4.2,4.4.3);
- Although our system did not outperform the IMDb performance (Sect.4.4.2), an in-deep evaluation of users preferences has shown that the average precision gained by LSA is very close to the precision of IMDb (Sect.4.4.3); it would therefore be necessary to conduct an online analysis of the behaviour of the system in order to better understand how it performs compared to IMDb.

Finally, we can not ignore that IMDb is strongly affected by user experiences: it uses features such as user votes, genre, title, keywords, and, most importantly, user recommendations themselves to generate an automatic response. On the contrary, our content-based recommendations system is user independent. Thus, our system can be also used to make recommendations when knowledge of users preferences is not available.

5 RELATED WORK

Recommendation algorithms are usually classified in content-based and collaborative filtering (Ekstrand

et al., 2011). Collaborative filtering systems are widely industrially utilized, for example by Amazon, MovieLens and Netflix, and recommendation is computed by analysing user profiles and user ratings of the items. When user preferences are not available, as in the start-up phase, or not accessible, due to privacy issues, it might be necessary to develop a content-based recommendation algorithm, or combined different approaches as in hybrid systems.

Among recommendation systems (Adomavicius and Tuzhilin, 2005), content-based recommendation systems rely on item descriptions that usually consist of punctual data.

Jinni¹⁸ is a movie recommendation system that analyses as well movie plots, but, differently from our approach, relies on user ratings, manual annotations and machine learning techniques.

LSA was shown to perform better than the simpler word and n-gram feature vectors in an interesting study (Lee and Welsh, 2005) where several types of vector similarity metrics (e.g., binary vs. count vectors, Jaccard vs. cosine vs. overlap distance measure, etc.) have been evaluated and compared.

Due to the high computational cost of LSA there have been many work around in the area of approximate matrix factorization; these algorithms maintain the spirit of SVD but are much easier to compute (Koren et al., 2009). For example, in (Gemulla et al., 2011) an effective distributed factorization algorithm based on stochastic gradient descent is shown. We opted for a scalable implementation of the process that does not require the term-document matrix to be stored in memory and is therefore independent of the corpus size (Řehůřek and Sojka, 2010).

Also the LDA Topic Model has been already ap-

¹⁸<http://www.jinni.com/>

plied in recommendation systems to analyze textual information. In particular in (Jin et al., 2005) a Web recommendation system to help users in locating information on the Web is proposed. In this system LDA is used as technique for discovering hidden semantic relationships among Web items by analyzing their content information. Another interesting application is described in (Krestel et al., 2009) where the authors propose a tag recommendation system where LDA is exploited to suggest tags for new resources.

In the specific domain of movie recommendation systems, we found only few frameworks that make use of plots. In particular in (Shi et al., 2013) a Context-Aware Recommendation algorithm is introduced, the framework combines the similarity based on plot keywords with a mood-specific movie similarity for providing recommendations. Also in (Moshfeghi et al., 2011) authors attempts to solve the cold start problem (where there is no past rating for an item) for collaborative filtering recommendation systems. The paper describes a framework, based on an extended version of LDA, able to take into account item-related emotions, extracted from the movie plots, and semantic data, inferred from movie features.

6 CONCLUSIONS AND FUTURE WORK

The paper presented a plot-based recommendation system. The system classifies two videos as being similar if their plots are alike. Two Topic Models, LDA and LSA, have been implemented and integrated within the recommendation system. The techniques have been compared and tested over a large collection of movies. The local movie database MongoDB has been created to store a large amount of metadata related to multimedia content coming from different sources with heterogeneous schemata.

Experimental evaluation of both LDA and LSA has been conducted to provide answers in term of efficiency and effectiveness. LSA turns out to be superior to LDA. The performance of both the techniques have been compared to user evaluation, and commercial approaches. LSA has been revealed to be better than LDA in supporting the suggestion of similar plots, however it does not outperform the commercial approach (IMDb). However, it is important to notice that our system does not rely on human effort and can be ported to any domain where natural language descriptions exist. Moreover, a nice feature of the system is its independence from the movie ratings expressed by users; thanks to this independence, it is

able to propose famous and beloved movies as well as old or unheard movies/programs that are similar to the content of the video the user has watched. This allows the system to find strongly related movies that other recommendation systems, such as IMDb, do not consider, as they has a low number of ratings.

The results shown in this paper highlight some limitations and stimulate some future directions for our research.

The plot-based recommendation techniques assume that the main feature a user likes in a movie is the plot, i.e. the content of the movie, if this is not the case, the system will fail in suggestion similar movies. Thus, we should couple our recommendation system with other techniques that do not totally rely on the plot.

Another major limitation is the description, i.e. the content of the plot. Most of the entries in the DBpedia collection do not have a plot. Even within a database like IMDb (the most accurate) not all the plots are described in a similar manner. Some are described by only one or two sentences, others, instead, are meticulously detailed. In addition to this, in most of the movies, the plot is not completely revealed, and this leads to have several movie descriptions that are partial.

While LDA deals with polysemy issue, LSA does not. This problem can be faced by making use of a lexical database as WordNet¹⁹. Each keyword might be replaced by its meaning (synset), before the application of the weight techniques. To understand which of the synsets better express the meaning of a keyword in a plot we may adopt Word Sense Disambiguation techniques (Navigli, 2009). The semantic relationships between synsets can be used for enhancing the keyword meaning by adding all its hypernyms and hyponyms (Po and Sorrentino, 2011; Sorrentino et al., 2010).

We used our system to compute the similarity of a movie with other movies in the database. However, in general we can use it to evaluate the similarity of textual descriptions such as plots of movies not present in the DB or news, book plots, book reviews etc.

For example, the system can find movies that contain a story similar to the one tell in a book, e.g. a movie or a television series that used it as a script, or dramatic movies based on true events similar to a news. The database could be expanded with other contents to suggest further items similar to the selected movie (e.g. if I liked a movie about the war in Cambodia I should be interested in newspaper articles, essays, or books about that topic).

¹⁹<http://wordnet.princeton.edu/>

ACKNOWLEDGEMENTS

We want to express our gratitude to Tania Farinella, Matteo Abbruzzo and Olga Kryukova, master students in Computer Engineering and Science at the Department of Engineering “Enzo Ferrari” at University of Modena and Reggio Emilia for their contribution in term of implementation of the first and second version of the system (without and with LDA) and for their support during the evaluation of the system.

Particular appreciation goes to Thomas Werner and Andreas Lahr²⁰, founders of vfree.it, for their suggestions and valuable comments on the paper.

REFERENCES

- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734–749.
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.
- Debnath, S., Ganguly, N., and Mitra, P. (2008). Feature weighting in content based recommendation system using social network analysis. In *Proceedings of the 17th international conference on World Wide Web, WWW '08*, pages 1041–1042, New York, NY, USA. ACM.
- Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.
- Dumais, S. T. (2004). Latent semantic analysis. *Annual Review of Information Science and Technology*, 38(1):188–230.
- Ekstrand, M. D., Riedl, J. T., and Konstan, J. A. (2011). Collaborative filtering recommender systems. *Found. Trends Hum.-Comput. Interact.*, 4(2):81–173.
- Farinella, T., Bergamaschi, S., and Po, L. (2012). A non-intrusive movie recommendation system. In *OTM Conferences (2)*, pages 736–751.
- Gemulla, R., Nijkamp, E., Haas, P. J., and Sismanis, Y. (2011). Large-scale matrix factorization with distributed stochastic gradient descent. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11*, pages 69–77, New York, NY, USA. ACM.
- Griffiths, T., Steyvers, M., and Tenenbaum, J. (2007). Topics in semantic representation. *Psychological Review*, 114(2):211–244.
- Gunawardana, A. and Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *The Journal of Machine Learning Research*, 10:2935–2962.
- Jin, X., Mobasher, B., and Zhou, Y. (2005). A web recommendation system based on maximum entropy. In *ITCC (1)*, pages 213–218. IEEE Computer Society.
- Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.
- Krestel, R., Fankhauser, P., and Nejdl, W. (2009). Latent dirichlet allocation for tag recommendation. In Bergman, L. D., Tuzhilin, A., Burke, R. D., Felfernig, A., and Schmidt-Thieme, L., editors, *RecSys*, pages 61–68. ACM.
- Lee, M. D. and Welsh, M. (2005). An empirical evaluation of models of text document similarity. In *Proceedings of the 27th Annual Conference of the Cognitive Science Society, CogSci2005*, pages 1254–1259. Erlbaum.
- Moshfeghi, Y., Piwowarski, B., and Jose, J. M. (2011). Handling data sparsity in collaborative filtering using emotion and semantic based features. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11*, pages 625–634, New York, NY, USA. ACM.
- Musto, C. (2010). Enhanced vector space models for content-based recommender systems. In *Proceedings of the Fourth ACM Conference on Recommender Systems, RecSys '10*, pages 361–364, New York, NY, USA. ACM.
- Navigli, R. (2009). Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2).
- Park, L. A. F. and Ramamohanarao, K. (2009). An analysis of latent semantic term self-correlation. *ACM Trans. Inf. Syst.*, 27(2):8:1–8:35.
- Po, L. and Sorrentino, S. (2011). Automatic generation of probabilistic relationships for improving schema matching. *Inf. Syst.*, 36(2):192–208.
- Rashid, A. M., Karypis, G., and Riedl, J. (2008). Learning preferences of new users in recommender systems: an information theoretic approach. *SIGKDD Explor. Newsl.*, 10(2):90–100.
- Řehůřek, R. and Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. <http://is.muni.cz/publication/884893/en>.
- Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18:613–620.
- Shi, Y., Larson, M., and Hanjalic, A. (2013). Mining contextual movie similarity with matrix factorization for context-aware recommendation. *ACM Trans. Intell. Syst. Technol.*, 4(1):16:1–16:19.
- Sorrentino, S., Bergamaschi, S., Gawinecki, M., and Po, L. (2010). Schema label normalization for improving schema matching. *Data Knowl. Eng.*, 69(12):1254–1273.

²⁰thomas.werner@vfree.tv, andreas.lahr@vfree.tv

Product Feature Taxonomy Learning based on User Reviews

Nan Tian¹, Yue Xu¹, Yuefeng Li¹, Ahmad Abdel-Hafez¹ and Audun Josang²

¹Faculty of Science and Engineering, Queensland University of Technology, Brisbane, Australia

²Department of Informatics, University of Oslo, Oslo, Norway

{n.tian, yue.xu, y2.li, a.abdelhafez}@qut.edu.au, josang@mn.uio.no

Keywords: Feature Extraction, Opinion Mining, Association Rules, Feature Taxonomy, User Reviews.

Abstract: In recent years, the Web 2.0 has provided considerable facilities for people to create, share and exchange information and ideas. Upon this, the user generated content, such as reviews, has exploded. Such data provide a rich source to exploit in order to identify the information associated with specific reviewed items. Opinion mining has been widely used to identify the significant features of items (e.g., cameras) based upon user reviews. Feature extraction is the most critical step to identify useful information from texts. Most existing approaches only find individual features about a product without revealing the structural relationships between the features which usually exist. In this paper, we propose an approach to extract features and feature relationships, represented as a tree structure called feature taxonomy, based on frequent patterns and associations between patterns derived from user reviews. The generated feature taxonomy profiles the product at multiple levels and provides more detailed information about the product. Our experiment results based on some popularly used review datasets show that our proposed approach is able to capture the product features and relations effectively.

1 INTRODUCTION

In recent years, the user generated online content exploded due to the advent of Web 2.0. For instance, online users write reviews to how they enjoy or dislike a product they purchased. This helps to identify features or characteristics of the product from users' point of view, which is an important addition to the product specification. However, to identify the relevant features from users' subjective review data is extremely challenging.

Feature-based opinion mining has attracted big attention recently. A significant amount of research has been proposed to improve the accuracy of feature generation for products (Hu and Liu, 2004a; Scaffidi et al., 2007; Hu et al., 2010; Zhang and Zhu, 2013; Popescu and Etzioni, 2005; Ding et al., 2008). However, most techniques only extract features; the structural relationship between product features has been omitted. For example, “*picture resolution*” is a common feature of digital camera in which “*resolution*” expresses the specific feature concept to describe the general feature “*picture*”. Yet, existing approaches treat “*resolution*” and “*picture*” as two individual features instead of finding the relationship between them. Thus, the information derived by existing feature extraction approaches is not sufficient for gen-

erating a precise product model since all features are allocated in the same level and independent from each other.

Association rule mining is a well explored method in data mining (Pasquier et al., 1999). Based on association rules generated from a collection of item transactions, we can discover the relations between items. However, the amount of generated association rules is usually huge and selecting the most useful rules is challenging (Xu et al., 2011). In our research, we propose to identify a group of frequent patterns as potential features to assist selecting useful association rules. The selected rules are used to identify relationships between features. Furthermore, in order to ensure that the most useful rules are to be selected, we also propose to apply statistical topic modelling technique (Blei et al., 2003) to the selection of association rules.

Our approach takes advantages of existing feature extraction approaches and makes two contributions. Firstly, we present a method to make use of association rules to find related features. Secondly, we create a product model called feature taxonomy which represents the product more accurately by explicitly representing the concrete relationships between general features and specific features.

2 RELATED WORK

Our research aims to extract useful product information based on user generated information to create a product model. This work is closely related to feature-based opinion mining which has drawn many researchers' attention in recent years. In detail, identifying features that have been mentioned by users is considered the most significant step in opinion mining (Hai et al., 2013). Hu and Liu (2004) first proposed a feature-based opinion mining method to extract features and sentiments from customer reviews. They use pattern mining to find frequent itemsets (nouns). These itemsets are pruned and considered frequent product features. A list of sentiment words (adjectives) that are nearby frequent features in reviews can be extracted and used to identify those product features that cannot be identified by pattern mining. Scaffidi et al. (2007) improved the performance of feature extraction in their proposed system called Red Opal. Specifically, they made use of a language model to find features by comparing the frequency of nouns in the review and in common use of English. Those frequent nouns in both reviews and in common use are considered invalid features. Hu et al. (2010) make use of SentiWordNet to identify all sentences that may contain users' sentiment polarity. Then, the pattern mining is applied to generate explicit features based on these opinionated sentences. In addition, a mapping database has been constructed to find those implicit features represented by sentiment words (e.g., *expensive* indicates *price*). To enhance the accuracy of finding correct features from free text review, Hai et al (2013) proposed a novel method which evaluates the domain relevance of a feature by exploiting features' distribution disparities across different corpora (domain-dependent review corpus such as cellphone reviews and domain-irrelevant corpus such as culture article collection). In detail, the *intrinsic-domain relevance* (IDR) and *extrinsic-domain relevance* (EDR) have been proposed to benchmark if a examined feature is related to a certain domain. The candidate feature with low IDR and high EDR scores will be pruned.

Lau et al. (2009) presented an ontology-based approach to profile the product. In detail, a number of ontology levels, such as feature level that contains identified features for a certain product and sentiment level in which sentiment words that describe a certain feature are stored, have been constructed (Lau et al., 2009). This method provides a simple product profile rather than extracting product features only.

The statistical topic modeling technique has been used in various fields such as text mining (Blei et al.,

2003; Hofmann, 2001) in recent years. Latent Semantic Analysis (LSA) is first proposed to capture the most significant features of a document collection based upon semantic structure of relevant documents (Lewis, 1992). Then, Probabilistic LSA (pLSA) (Hofmann, 2001) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) are proposed to improve the interpretation of results from LSA. These techniques have been proven more effective on document modeling and topic extraction, which are represented by topic-document and word-topic distribution, respectively. Particularly, multinomial distribution over words which is derived based upon word frequency can be generated to represent topics in a given text collection.

None of aforementioned feature identification approaches is able to identify the relationships between the extracted product features. The structural relationships that exist between features can be used to describe the reviewed product in more depth. However, how to evaluate and determine the relations between features is still challenging.

The remainder of the paper is organized as follows. The next section illustrates the construction process of our proposed feature taxonomy. Then, the evaluation of our approach is reported afterwards. Finally, we conclude and describe future direction of our research work.

3 THE PROPOSED APPROACH

Our proposed approach consists of two main steps: product taxonomy construction using association rules and taxonomy expansion based on reference features. The input of our system is a collection of user reviews for a certain product. The output is a product feature taxonomy which contains not only all generated features but also the relationships between them.

3.1 Pre-processing and Transaction File Generation

First of all, we construct a single document called an *aggregated review document* which combines all the reviews in a collection of reviews, keeping each sentence in the original reviews as one sentence in the constructed *aggregated review document*. Three steps are undertaken to process the review text in order to extract useful information. Firstly, we generate the part-of-speech (POS) tag for each word in the *aggregated review document* to indicate whether the word is a *noun*, *adjective* or *adverb* etc. For instance, after the POS tagging, "*The flash is very weak.*" would

be transformed to “*The/DT flash/NN is/VBZ very/RB weak/JJ .I.*”, where *DT*, *NN*, *VBZ*, *RB*, and *JJ* represent Determiner, Noun, Verb, Adverb and Adjective, respectively. Secondly, according to the thumb rule that most product features are nouns or noun phrases (Hu and Liu, 2004b), we process each sentence in the *aggregated review document* to only keep words that are nouns. All the remaining nouns are also pre-processed by stemming and spelling correction. Each sentence in the *aggregated review document* consists of all identified nouns of a sentence in the original reviews. Finally, a transactional dataset is generated from the *aggregated review document*. Each sentence which consists of a sequence of nouns in the *aggregated review document* is treated as a transaction in the transactional dataset.

3.2 Potential Features Generation

Our first task is to generate potential product features that are expressed by those identified nouns or noun phrases. According to (Hu and Liu, 2004a), significant product features are discussed extensively by users in reviews (e.g., “*battery*” for cameras). Upon this, most existing feature extraction approaches make use of pattern mining techniques to find potential features. Specifically, an itemset is a set of items (i.e., words in review text in this paper) that appear together in one or multiple transactions in a transactional dataset. Given a set of items, $I = \{i_1, i_2, \dots, i_n\}$, an itemset is defined as $X \subseteq I$. The support of an itemset X , denoted as $Supp(X)$, is the percentage of transactions in the dataset that contain X . All frequent itemsets from a set of transactions that satisfy a user-specified minimum support will be extracted as the potential features. However, not all frequent itemsets are genuine since some of them may be just frequent but meaningless. We use compactness pruning method proposed by (Hu and Liu, 2004a) to filter frequent itemsets. After the pruning, we can get a list of frequent itemsets that are considered potential features, denoted as *FP*.

3.3 Product Feature Taxonomy Construction

In this step, we propose to utilize association rules generated from the discovered potential product features to identify relations in order to construct a feature taxonomy.

Association rule mining can be described as follows: Let $I = \{i_1, i_2, \dots, i_n\}$, be a set of items, and the dataset consists of a set of transactions $D = \{t_1, t_2, \dots, t_m\}$. Each transaction t contains a subset of

items from I . Therefore, an association rule r represents an implication relationship between two itemsets which can be defined as the form $X \rightarrow Y$, where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The itemsets X and Y are called antecedent and consequent of the rule, respectively. To assist selecting useful rules, the support $Supp(X \cup Y)$ and the confidence $Conf(X \rightarrow Y)$ of the rule can be used (Xu et al., 2011).

For easily describing our approach, we define some useful and important concepts as follows:

Definition 1 (Feature Taxonomy): A feature taxonomy consists of a set of features and their relationships, denoted as $FH = \{F, L\}$, F is a set of features where $F = \{f_1, f_2, \dots, f_n\}$ and L is a set of relations. The feature taxonomy has the following constraints:

- (1) The relationship between a pair of features is the sub-feature relationship. For $f_i, f_j \in F$, if f_j is a sub feature of f_i , then (f_i, f_j) is a link in the taxonomy and $(f_i, f_j) \in L$, which indicates that f_j is more specific than f_i . f_i is called the parent feature of f_j and denoted as $P(f_j)$.
- (2) Except for the root, each feature has only one parent feature. This means that the taxonomy is structured as a tree.
- (3) The root of the taxonomy represents the product itself.

Definition 2 (Feature Existence):For a given feature taxonomy $FH = \{F, L\}$, let $W(g)$ represent a set of words that appear in a potential feature g , let $ES(g) = \{a_i | a_i \in 2^{W(g)}, a_i \in F\}$ contain all subsets of g which exist in the feature taxonomy, $ES(g)$ is called the existing subsets of g , if $\bigcup_{a_i \in ES(g)} W(a_i) = W(g)$, then g is considered exist in FH , denoted as $exist(g)$, otherwise $\neg exist(g)$.

Opinion mining is also referred as sentiment analysis (Subrahmanian and Reforgiato, 2008; Abbasi et al., 2008; Wright, 2009). Adjectives or adverbs that appear together with product features are considered as the sentiment words in opinion mining. The following definition defines the sentiment words that are related to a product feature.

Definition 3 (Related Sentiments): For a feature $f \in F$, let $RS(f)$ denote a set of sentiment words which appear in the same sentences as f in user reviews, $RS(f)$ is defined as the related sentiments of f .

Definition 4 (Sentiment Sharing): For features $f_1, f_2 \in F$, the sentiment sharing between f_1 and f_2 is defined as $SS(f_1, f_2) = |RS(f_1) \cap RS(f_2)|$.

For deriving sub features using association rules, we need to select a set of useful rules rather than using all the rules. In the next two subsections, we will first

propose two methods to select rules, one method is to select rules based on the sentiment sharing among features and the other method is to select rules by using the word relatedness derived from the results generated by using the typical topic model technique method LDA (Blei et al., 2003); then introduce some strategies to update the feature taxonomy by adding sub features using the selected rules.

In order to explain the topic modelling based method, we first define some related concepts. Let $RE = \{r_1, r_2, \dots, r_M\}$ be a collection of reviews, each review consists of nouns only, $W = \{w_1, w_2, \dots, w_n\}$ be a set of words appearing in RE , and $Z = \{Z_1, \dots, Z_v\}$ be a set of pre-specified hidden topics. LDA can be used to generate topic models for representing the collection as a whole and also for each review in the collection. At the collection level, the topic model represents the collection RE using a set of topics each of which is represented by a probability distribution over words (i.e., nouns in the context of this paper) for topic. In this paper, we will use the collection level representation to find the relatedness between words.

At collection level, each topic Z_j is represented by a probability distribution over words, $\phi_j = \{p(w_1|Z_j), p(w_2|Z_j), \dots, p(w_n|Z_j)\}$, $\sum_{k=1}^n \phi_{j,k} = 1$, $p(w_k|Z_j)$ is the probability of word w_k being used to represent the topic Z_j . Based on the probability $p(w_k|Z_j)$, we can choose the top words to represent the topic Z_j .

Definition 5 (Topic Words): Let $\phi_j = \{p(w_1|Z_j), p(w_2|Z_j), \dots, p(w_n|Z_j)\}$ be the topic representation for topic Z_j produced by LDA and $0 \leq \delta \leq 1$ be a threshold, a set of the topic words for Z_j , denoted as $TW(Z_j)$, is defined as $TW(Z_j) = \{w|w \in W, p(w|Z_j) > \delta\}$.

Definition 6 (Word Relatedness): We use word relatedness to indicate how likely that two words have been used to represent a topic together. Let $w_i, w_j \in W$ be two words, the word relatedness between two words with respect to topic z is defined below:

$$WR_z(w_i, w_j) = \begin{cases} 1 - |p(w_i|z) - p(w_j|z)| & w_i \in TW(z) \\ & \text{and } w_j \in TW(z) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Definition 7 (Feature Topic Representation): For feature $f \in F$, let $WD(f)$ be a set of words appearing in f and $TW(z)$ be the topic words of topic z . If $WD(f) \subset TW(z)$, the feature topic representation of feature f for topic z is defined as $FTP(f, z) = \{(w, p(w|z)) | w \in WD(f)\}$.

Definition 8 (Feature Relatedness): For features $f_i, f_j \in F$, if both features appear in a certain topic

z , then the feature relatedness between f_i and f_j with respect to z is defined as:

$$FR_z(f_i, f_j) = \min_{\substack{w_i \in WD(f_i) \\ w_j \in WD(f_j)}} \{WR_z(w_i, w_j)\} \quad (2)$$

3.3.1 Rule Selection

Let $R = \{r_1, r_2, \dots, r_n\}$ be a set of association rules generated from the frequent itemsets FP , each rule r in R has the form $X_r \rightarrow Y_r$, X_r and Y_r are the antecedent and consequent of r , respectively.

Assuming that f_e is a feature which has already been in the current feature taxonomy FH , to generate the sub features for f_e , we first select a set of candidate rules, denoted as $R_{f_e}^c$, which could be used to generate the sub features:

$$R_{f_e}^c = \{X \rightarrow Y | X \rightarrow Y \in R, X = f_e, \text{Supp}(X) > (Y)\} \quad (3)$$

As defined in Equation (3), the rules in $R_{f_e}^c$ should satisfy two constraints. The first constraint, $X = f_e$, specifies that the antecedent of a selected rule must be the same as the feature f_e . Sub features represent specific cases of a feature, they are more specific compared to the feature. The second constraint is based on the assumption that more frequent itemsets usually represent more general concepts, and less frequent itemsets usually represent more specific concepts. For instance, according to our observation toward features, a general feature (e.g., “*picture*”, its frequency is 62) appears more frequently than a specific feature (e.g., “*resolution*”, its frequency is 9) in reviews for the camera 2 in the dataset published by Liu (Ding et al., 2008). Therefore, only the rules which can derive more specific features will be selected.

However, not all selected rules represent correct sub-feature relationship. For instance, $mode \rightarrow auto$ is more appropriate for describing a sub-feature relationship rather than $camera \rightarrow auto$. Therefore, the rule $camera \rightarrow auto$ should not be considered when we generate the sub features for “*camera*”. Upon this, we aim to prune the unnecessary rules before generating sub features for each taxonomy feature. Firstly, a feature and its sub features should share similar sentiment words since they describe the same aspect of a product at different abstract levels (e.g., *vivid* can be use to describe both *picture* and *color*). Therefore, we should select rules whose antecedent (representing the feature) and consequent (representing a possible sub feature) share as many sentiment words as possible because the more sentiment words they share, the more possible they are about the same aspect of the product. Secondly, based on topic models

generated from LDA, the more a feature and its potential sub feature appear in the same topics, the more likely they are related to each other.

Let f_X, f_Y be two features and $Z_{(f_X, f_Y)}$ be a set of topics that contains both features, the feature relatedness between f_X, f_Y with respect to all topics, denoted as $FR_{avg}(f_X, f_Y)$, is defined as the average feature relatedness between the two features over $Z_{(f_X, f_Y)}$:

$$FR_{avg}(f_X, f_Y) = \frac{\sum_{z \in Z_{(f_X, f_Y)}} FR_z(f_X, f_Y)}{|Z_{(f_X, f_Y)}|} \quad (4)$$

Based on this view, we propose the following equation to calculate a score for each candidate rule $X \rightarrow Y$ in $R_{f_e}^c$:

$$\begin{aligned} Weigh(X \rightarrow Y) = & \alpha(Supp(Y) \times Conf(X \rightarrow Y)) + \\ & \beta \frac{SS(X, Y)}{|RS(X) \cup RS(Y)|} + \gamma FR_{avg}(X, Y) \end{aligned} \quad (5)$$

$0 < \alpha, \beta, \gamma < 1$. The value of α, β , and γ is set to 0.8, 0.1, and 0.1, respectively in our experiment described in Section 4. There are three parts in Equation (5). The first part is used to measure the belief to the consequent Y by using this rule since $Conf(X \rightarrow Y)$ measures the confidence to the association between X and Y and $Supp(Y)$ measures the popularity of Y . The second part is the percentage of the shared sentiment words given by $SS(X, Y)$ over all the sentiment words used for either X or Y . Yet, the third part in the equation is the average feature relatedness between X and Y . Given a threshold σ , we propose to use the following equation to select the rules from the candidate rules in $R_{f_e}^c$. The rules in $R_{f_e}^c$ will be used to derive sub features for the features in FP . $R_{f_e}^c$ is called the rule set of f_e .

$$\begin{aligned} R_{f_e} = & \{X \rightarrow Y | X \rightarrow Y \in R_{f_e}^c, \\ & Weigh(X \rightarrow Y) > \sigma\} \end{aligned} \quad (6)$$

3.3.2 Feature Taxonomy Construction

Let $FH = \{F, L\}$ be a feature taxonomy which could be an empty tree, FP be a set of frequent itemsets generated from user reviews which are potential features, and R be a set of rules generated from user reviews. This task is to construct a feature taxonomy if F is empty or update the feature taxonomy if F is not empty by using the rules in R . Let UF be a set of features on the tree which need to be processed in order to construct or update the tree. If F is empty, the itemset in FP which has the highest support will be chosen as the root of FH , it will be the only item

in UF at the beginning. If F is not empty, UF will be F , i.e., $UF = F$.

Without losing generality, assuming that F is not empty and the set of features currently on the tree, UF is the set of features which need to be processed to update or construct the tree. For each feature in UF , let f_e be a feature in UF , i.e., $f_e \in UF$ and $X \rightarrow Y \in R_{f_e}$ be a rule with $X = f_e$, the next step is to decide whether or not Y should be added to the feature taxonomy as a sub feature of f_e . There are two possible situations: Y does not exist in the feature taxonomy, i.e., $\neg exist(Y)$ and Y does exist in the taxonomy, i.e., $exist(Y)$. In the first situation, the feature taxonomy will be updated by adding Y as a sub feature of f_e , i.e., $F = F \cup \{Y\}$, $L = L \cup (f_e, Y)$, and Y should be added to UF for further checking.

In the second situation, i.e., Y already exists in the taxonomy, i.e., according to Definition 2, there are two cases, $Y \notin ES(Y)$ (i.e., Y is not in the tree) or $Y \in ES(Y)$ (i.e., Y is in the tree). In the first case, Y is not considered a sub feature of f_e and consequently, no change is required to the tree. In the second case, $\exists f_y \in F$, f_y is the parent feature of Y , i.e., $P(Y) = f_y$ and $(f_y, Y) \in L$. Now, we need to determine whether to keep f_y as the parent feature of Y or change the parent feature of Y to f_e . That is, we need to examine f_y and f_e to see which of them is more suitable to be the parent feature of Y . The basic strategy is to compare f_y and f_e to see which of them has more sentiment sharing and feature relatedness with Y . Let f_p, f_c be a potential parent feature and sub feature, respectively. We propose a ranking equation to indicate how likely f_c is related to f_p : $Q(f_p, f_c) = \frac{SS(f_p, f_c)}{RS(f_c)} + FR_{avg}(f_p, f_c)$. Thus, if $Q(f_y, Y) < Q(f_e, Y)$, the link (f_y, Y) will be removed from the taxonomy tree, (f_e, Y) will be added to the tree, otherwise, no change to the tree and f_y is still the parent feature of Y .

3.3.3 Algorithms

The construction of the feature taxonomy is to generate a feature tree by finding all sub features for each feature. In this section, we will describe the algorithms to construct the feature taxonomy. As mentioned above, if the tree is empty, the feature with the highest support will be chosen as the root. So, at the very beginning, F and UF contain at least one item which is the root. Algorithm 1 describes the method to construct or update a feature taxonomy.

After the taxonomy construction, some potential features may be left over in RF and have not been added to the taxonomy. The main reason is because these itemsets may not frequently occur in the reviews

Algorithm 1: Feature Taxonomy Construction.**Input:**
 $R, FH = \{F, L\}, FP.$
Output:
 FH, RF // RF is the remaining features which are not added to FH after the construction

- 1: if $F = \emptyset$, then $root := \operatorname{argmax}_{f \in FP} \{supp(f)\}$,
 $F := UF := \{root\}$;
- 2: else $UF := F$;
- 3: for each feature $f_e \in UF$
- 4: if $R_{f_e} \neq \emptyset$ //the rule set of f_e is not empty
- 5: for each rule $X \rightarrow Y \in R_{f_e}$
- 6: if $\neg exist(Y)$ // Y does not exist on the tree
- 7: $F := F \cup \{Y\}, L := L \cup (f_e, Y)$,
 $UF := UF \cup \{Y\}, FP := FP - \{Y\}$;
- 8: else // Y exists on the tree
- 9: if $Y \in ES(Y)$ and $Q(f_y, Y) < Q(f_e, Y)$
// f_y is Y 's parent feature
- 10: $L := L \cup (f_e, Y), L := L - (f_y, Y)$;
//add (f_e, Y) and remove (f_y, Y)
- 11: else // $Y \notin ES(Y)$, Y is not on the tree
- 12: $FP := FP - \{Y\}$;
- 13: endfor
- 14: endif
- 15: $UF := UF - \{f_e\}$; //remove f_e from UF
- 16: endfor
- 17: $RF := FP$

together with the features that have been added in the taxonomy. In order to prevent valid features from being missed out, we check those remaining itemsets in RF by examining the shared sentiment words and feature relatedness between the remaining itemsets and the features in the taxonomy. Let $FH = \{F, L\}$ be the constructed feature taxonomy, RF be the set of remaining potential features, for a potential feature g in RF , the basic strategy to determine whether g is a feature or not is to examine the Q ranking between g and the features in the taxonomy. Let $F_g = \{f | f \in F, Q(f, g) > 0\}$ be a set of features which are related to g , if $F_g \neq \emptyset$, g is considered a feature. The most related feature is defined as $f_m = \operatorname{argmax}_{f \in F_g} \{Q(f, g)\}$. g will be added to the taxonomy with f_m as its parent feature. If there are multiple such features f_m which have the highest ranking score with g , the one with the highest support will be chosen as the parent feature of g .

Algorithm 2 formally describes the method mentioned above to expand the taxonomy by adding the remaining features.

After the expansion, the features left over in RF are not considered as features for this product.

Algorithm 2: Feature Taxonomy Expansion.**Input:**
 $FH = \{F, L\}, RF.$
Output:
 FH

- 1: for each feature $g \in RF$
- 2: if $(F_g := \{f | f \in F, Q(f, g) > 0\}) \neq \emptyset$
- 3: $M := \{a | a \in F_g \text{ and}$
 $Q(a, g) = \max_{f \in F_g} \{Q(f, g)\}\}$
- 4: $f_m := \operatorname{argmax}_{f \in M} \{supp(f)\}$
- 5: $F := F \cup \{g\}, L := L \cup (f_m, g)$
- 6: $RF := RF - \{g\}$

4 EXPERIMENT AND EVALUATION

We use three datasets in the experiments. Each dataset contains user reviews for a certain type of digital cameras. One dataset is used in (Hu and Liu, 2004a), while the other two are used in (Ding et al., 2008). Each review in the datasets has been manually annotated. In detail, a human examiner read a review sentence by sentence. If a sentence is considered indicating the user's opinions, such as positive and negative, all possible features in the sentence that are modified by sentiment words are tagged. We take these annotated features as the correct features to evaluate the performance of our proposed method in feature extraction. The number of reviews and number of annotated features are 51 and 98 for camera 1, 34 and 75 for camera 2, and 45 and 105 for camera 3.

Our proposed feature taxonomy captures both product features and relations between features. Therefore, the evaluations are twofold: feature extraction evaluation and structural relations evaluation.

4.1 Feature Extraction Evaluation

First of all, we evaluate the performance of our approach by examining the number of accurate features in user reviews that have been extracted. We use the feature extraction method (FBS) proposed in (Hu and Liu, 2004a) as the baseline for comparison. In addition, in order to examine the effectiveness of using the sentiment sharing measure, the feature relatedness measure, and the combination of the two, we conduct our experiment in four runs:

- (1) *Rule*: construct the feature taxonomy by only utilizing the information of association rules (i.e., support and confidence value only) without using the sentiment sharing and the feature relatedness measures;

- (2) *SS*: construct the feature taxonomy by taking the information of association rules and the sentiment sharing measure without using the feature relatedness measure;
- (3) *FR*: construct the feature taxonomy by taking the information of association rules and the feature relatedness measure without using the sentiment sharing measure;
- (4) *Hybrid*: the sentiment sharing and the feature relatedness are combined together with the information of association rules to construct the feature taxonomy.

Table 1: Recall Comparison.

| | Camera 1 | Camera 2 | Camera 3 | Average |
|--------|----------|----------|----------|---------|
| FBS | 0.57 | 0.63 | 0.57 | 0.59 |
| Rule | 0.38 | 0.52 | 0.45 | 0.45 |
| SS | 0.56 | 0.65 | 0.58 | 0.60 |
| FR | 0.56 | 0.67 | 0.58 | 0.60 |
| Hybrid | 0.56 | 0.68 | 0.58 | 0.61 |

Table 2: Precision Comparison.

| | Camera 1 | Camera 2 | Camera 3 | Average |
|--------|----------|----------|----------|---------|
| FBS | 0.45 | 0.42 | 0.51 | 0.46 |
| Rule | 0.55 | 0.57 | 0.74 | 0.62 |
| SS | 0.62 | 0.57 | 0.63 | 0.61 |
| FR | 0.60 | 0.56 | 0.63 | 0.60 |
| Hybrid | 0.62 | 0.59 | 0.68 | 0.63 |

Table 3: F1 Score Comparison.

| | Camera 1 | Camera 2 | Camera 3 | Average |
|--------|----------|----------|----------|---------|
| FBS | 0.50 | 0.50 | 0.54 | 0.51 |
| Rule | 0.45 | 0.54 | 0.56 | 0.52 |
| SS | 0.59 | 0.61 | 0.60 | 0.60 |
| FR | 0.58 | 0.61 | 0.60 | 0.60 |
| Hybrid | 0.59 | 0.63 | 0.63 | 0.62 |

Table 1, 2, 3 illustrate the recall, precision, and F1 score results produced in the four runs, respectively. From the results, we can see that using both the sentiment sharing and feature relatedness can obtain better feature extraction performance than the use of association rule's information only. In particular, the hybrid method, which uses both sentiment sharing and feature relatedness, achieves the best results in most cases. However, the size of the review dataset and the number of annotated features can affect the precision and recall, which makes the values of the precision and recall vary in different range for different datasets. For instance, camera 3 has higher precision values than camera 2 due to more reviews in camera 3 dataset than that in camera 2 dataset, but camera 3 has lower recall values than camera 2 due to more manually annotated features in camera 3 dataset.

4.2 Structural Relation Evaluation

The evaluation of the relations requires the standard taxonomy or knowledge from experts (Tang et al.,

2009). Since there is no existing standard taxonomy available for comparison, we manually created taxonomy for the three cameras according to the product technical specifications provided online by manufacture organizations^{1, 2, 3}. From the product specifications on these websites, each camera has a number of attributes such as *lens system* and *shooting modes*. In addition, each attribute may also have several sub attributes. For instance, the *shooting modes* of the camera contains more specific attributes (e.g., *intelligent auto* and *custom*). Based upon such information, we create the product feature taxonomy for three digital cameras and use the taxonomy as the testing taxonomy, called Manual Feature Taxonomy (*MFT*), to evaluate the relations within our proposed feature taxonomy.

Due to the difference between the technical specifications from domain experts and the subjective reviews from online users, the words used to represent a feature in user reviews are very often different from the words for the same feature specified by domain experts in the product specification. For example, the feature *lens system* in the testing taxonomy and the feature *lens* in our generated taxonomy should be the same according to common knowledge even though they are not exactly matched with each other. Because of this fact, we will determine the match between two features based on overlapping of the two features rather than exact matching.

Let $MFT = \{F_{MFT}, L_{MFT}\}$ be the testing taxonomy with F_{MFT} being a set of standard features given by domain experts and L_{MFT} being a set of links in the testing taxonomy. For a given link $(f_{Fp}, f_{Fc}) \in L$ in the constructed product feature taxonomy and two features $f_{Mp}, f_{Mc} \in F_{MFT}$ in the testing taxonomy, the link (f_{Fp}, f_{Fc}) is considered matched with (f_{Mp}, f_{Mc}) and therefore represent a correct feature relation if the following conditions are satisfied:

1. $W(f_{Mp}) \cap W(f_{Fp}) \neq \emptyset$ and $W(f_{Mc}) \cap W(f_{Fc}) \neq \emptyset$
2. There exists a path in MFT , $\langle f_{Mp}, f_1, f_2, \dots, f_n, f_{Mc} \rangle$, (f_{Mp}, f_1) , (f_i, f_{i+1}) , $(f_n, f_{Mc}) \in L_{MFT}$, $i = 1, \dots, n - 1$

We examine the testing taxonomy and the constructed taxonomy to identify all matched links in the constructed taxonomy. The traditional measures precision and recall are used to evaluate the correctness of the feature relations in the constructed feature taxonomy. Let $ML(FH)$ denote the matched links in

¹<http://www.canon.com.au/Personal/Products/Camerasand-Accessories/Digital-Cameras/PowerShot-S100>

²<http://www.nikonusa.com/en/Nikon-Products/Product/Compact-Digital-Cameras/26332/COOLPIX-S4300.html>

³http://www.usa.canon.com/cusa/support/consumer/digital_cameras/powershot_g_series/powershot_g3#Specifications

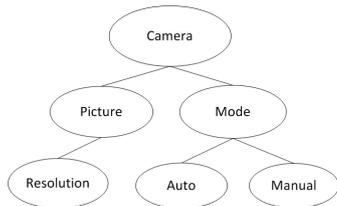


Figure 1: Constructed Feature Taxonomy.

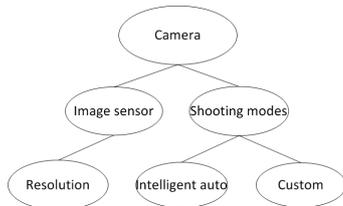


Figure 2: Testing Feature Taxonomy.

the constructed taxonomy, the precision and recall are defined as : $\text{Precision} = \frac{ML(FH)}{|L|}$ and $\text{Recall} = \frac{ML(FH)}{|L_{MFT}|}$.

Table 4: Recall and Precision of Relation Evaluation

| | Relations in MFT | Relations in FH | Recall | Precision |
|----------|------------------|-----------------|--------|-----------|
| Camera 1 | 75 | 97 | 0.40 | 0.46 |
| Camera 2 | 63 | 97 | 0.57 | 0.65 |
| Camera 3 | 71 | 102 | 0.51 | 0.57 |

Table 4 illustrates the evaluation results including the number of relations within the testing taxonomy, the number of relations within our generated taxonomy, recall and precision for the three different cameras, respectively. From the results, we can see that our generated feature taxonomy correctly capture around 50% of the relationships. Figure 1 and Figure 2 show a part of the feature taxonomy generated from our proposed approach and the testing taxonomy generated based on the product specification available online given by domain experts, respectively. From the comparison, our generated feature taxonomy identifies the relation between *picture* and *resolution*. Although the testing taxonomy uses more technical terms, which are *image sensor* instead of *picture*; in fact, they refer to the same attribute of the camera according to common knowledge. Similarly, the *(mode, auto)* and *(shooting modes, intelligent auto)* indicate the same relationship between two features.

As aforementioned, the online users and manufacture experts may describe the same feature by using totally different terms or words. This does affect the performance (both recall and precision) of our proposed approach in feature relationship identification negatively. For instance, the user may prefer using “*manual*” to depict a specific camera mode option. By contrast, the manufacture experts usually pick the term “*custom*” to describe this sub feature which be-

longs to “*shooting modes*”. In such a case, the two relations: *(mode, manual)* and *(shooting modes, custom)* cannot match.

5 CONCLUSION AND FUTURE WORK

In this paper, we introduced a product feature taxonomy learning approach based on frequent patterns and association rules. The objective is to not only extract product features mentioned in user reviews but also identify the relationship between the generated features. The results of our experiment indicate that our proposed approach is effective in both identifying correct features and structural relationship between them. Particularly, the feature relationships captured in the feature taxonomy provide more detailed information about products. This leads us to represent products profiles as multi-levels of feature, rather than a single level as most other methods do.

In the future, we plan to improve and evaluate our proposed product model by utilizing semantic similarity tools. For instance, the vocabulary mismatch can be handled by examining the semantic similarity when we undertake the structural relation evaluation. In addition, we plan to develop a review recommender system that makes use of the proposed product model in order to identify high quality reviews. The structural relations of the product model are able to assist identifying some characteristics of reviews, such as how a certain feature and its sub features have been discussed and how many different features have been covered. Our system will therefore aim at recommending reviews based upon such criteria to help users make purchasing decisions.

REFERENCES

- Abbasi, A., Chen, H., and Salem, A. (2008). Sentiment analysis in multiple languages: Feature selection for opinion classification in web forum. *ACM Transactions on Information Systems*, 26(3).
- Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993 – 1022.
- Ding, X., Liu, B., and Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 231 – 240.
- Hai, Z., Chang, K., Kim, J., and Yang, C. (2013). Identifying features in opinion mining via intrinsic and extrinsic domain relevance. *IEEE Transactions on Knowledge and Data Engineering*, pages 1 – 1.

- Hofmann, T. (2001). Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1 - 2):177 – 196.
- Hu, M. and Liu, B. (2004a). Mining and summarizing customer reviews. In *10th ACM SIGKDD international conference on Knowledge discovery and data mining*.
- Hu, M. and Liu, B. (2004b). Mining opinion features in customer reviews. In *Proceedings of the 19th national conference on Artificial intelligence*.
- Hu, W., Gong, Z., and Guo, J. (2010). Mining product features from online reviews. In *IEEE International Conference on E-Business Engineering*, pages 24 – 29.
- Lau, R. Y., Lai, C. C., Ma, J., and Li, Y. (2009). Automatic domain ontology extraction for context-sensitive opinion mining. In *Proceedings of the Thirtieth International Conference on Information Systems*.
- Lewis, D. D. (1992). An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th ACM International Conference on Research and Development in Information Retrieval*, pages 177 – 196.
- Pasquier, N., Bastide, Y., Taouil, R., and Lakhal, L. (1999). Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24(1):25 – 46.
- Popescu, A.-M. and Etzioni, O. (2005). Extracting product features and opinions from reviews. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 339–346.
- Scaffidi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H., and Jin, C. (2007). Red opal: Product-feature scoring from reviews. In *Proceedings of the 8th ACM conference on Electronic commerce*, number 182 - 191.
- Subrahmanian, V. S. and Reforgiato, D. (2008). Ava: Adjective-verb-adverb combinations for sentiment analysis. *IEEE Intelligent Systems*, pages 43 – 50.
- Tang, J., Leung, H.-f., Luo, Q., Chen, D., and Gong, J. (2009). Towards ontology learning from folksonomies. In *Proceedings of the 21st international joint conference on Artificial intelligence*, pages 2089 – 2094.
- Wright, A. (2009). Our sentiments, exactly. *Communications of the ACM*, 52(4):14 – 15.
- Xu, Y., Li, Y., and Shaw, G. (2011). Representations for association rules. *Data and Knowledge Engineering*, 70(6):237 – 256.
- Zhang, Y. and Zhu, W. (2013). Extracting implicit features in online customer reviews for opinion mining. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 103 – 104.

Automatic Web Page Classification Using Visual Content

António Videira and Nuno Gonçalves

Institute of Systems and Robotics, University of Coimbra, Coimbra, Portugal
{avideira, nunogon}@isr.uc.pt

Keywords: Web Page Classification, Feature Extraction, Feature Selection, Machine Learning.

Abstract: There is a constantly increasing requirement for automatic classification techniques with greater classification accuracy. To automatically classify and process web pages, the current systems use the text content of those pages. However, little work has been done on using the visual content of a web page. On this account, our work is focused on performing web page classification using only their visual content. First a descriptor is constructed, by extracting different features from each page. The features used are the simple color and edge histograms, Gabor and Tamura features. Then two methods of feature selection, one based on the Chi-Square criterion, the other on the Principal Components Analysis are applied to that descriptor, to select the top discriminative attributes. Another approach involves using the Bag of Words (BoW) model to treat the SIFT local features extracted from each image as words, allowing to construct a dictionary. Then we classify web pages based on their aesthetic value, their recency and type of content. The machine learning methods used in this work are the Naïve Bayes, Support Vector Machine, Decision Tree and AdaBoost. Different tests are performed to evaluate the performance of each classifier. Finally, we thus prove that the visual appearance of a web page has rich content not explored by current web crawlers based only on text content.

1 INTRODUCTION

Over the last years, the world has witnessed a huge growth on the internet, with millions of web pages on every topic easily accessible through the web, making the web a huge repository of information. Hence there is need for categorizing web documents to facilitate the indexing, searching and retrieving of pages. In order to achieve web's full potential as an information resource, the vast amount of content available in the internet has to be well described and organized. That is why automation of web page classification (WPC) is useful. WPC helps in focused crawling, assists in the development and expanding of web directories (for instance Yahoo), helps in the analysis of specific web link topic, in the analysis of the content structure of the web, improves the quality of web search (e.g., categories view, ranking view), web content filtering, assisted web browsing and much more.

Since the first websites in the early 1990's, designers have been innovating the way websites look. The visual appearance of a web page influences the way the user will interact with it. The structural elements of a web page (e.g. text blocks, tables, links, images) and visual characteristics (e.g., color, size) are used to determine the visual presentation and level of complexity of a page. This visual presentation is known

as Look and Feel, which is one of the most important properties of a web page. The visual appearance (Look and Feel) of each website is constructed using colors and color combinations, type fonts, images and videos, and much more.

The aim of this work is to enable automatic analysis of this visual appearance of web pages by using the web page as it appears to the user and evaluate the performance of different classifiers in the classification of web pages in several tasks.

The motivation behind our work is based on (de Boer et al., 2010), where the authors proved that by using generic visual features it was possible to classify web pages for several different types of tasks. They classify web pages based on their aesthetic value, their design recency and the type of website. They concluded that by using low-level features of web pages, it is possible to distinguish between several classes that vary in their Look and Feel, in particular aesthetically well designed vs. badly designed, recent vs. old fashioned and different topics. We extend their work by using and comparing several features, testing new feature selection methods and classifiers. We used the same binary variables (aesthetic value and design recency) but extended the type of webpage content for 8 classes instead of 4. We also aim to obtain better accuracy in classification.

2 RELATED WORK

The text content that is directly located on the page is the most used feature. A WPC method presented by Selamat and Omatu (Selamat and Omatu, 2004) used a neural network with inputs based on the Principal Component Analysis and class profile-based features. By selecting the most regular words in each class and weighted them, and with several methods of classification, they were able to demonstrate an acceptable accuracy. Chen and Hsieh (Chen and Hsieh, 2006) proposed a WPC method using a SVM based on a weighted voting scheme. This method uses Latent semantic analysis to find relations between keywords and documents, and text features extracted from the web page content. Those two features are then sent to the SVM model for training and testing respectively. Then, based on the SVM output, a voting scheme is used to determine the category of the web page.

There are few studies of WPC using the visual content, because traditionally only text information is used, achieving reasonable accuracy. It has been, however, noticed (de Boer et al., 2010) that the visual content can help in disambiguating the classification based only on this text content. Additionally, another factor in favor of using the visual content is the fact that subjective variables as design recency and aesthetic value cannot be studied using text content contained in the html code. These variables are increasing in importance due to web marketing strategies.

A WPC approach based on the visual information was implemented by Asirvatham et al. (Asirvatham and Ravi, 2001), where a number of visual features, as well as text features, were used. They proposed a method for automatic categorization of web pages into a few broad categories based on the structure of the web documents and the images presented on it. Another approach was proposed by Kovacevic et al. (Kovacevic et al., 2004), where a page is represented as a hierarchical structure - Visual Adjacency Multi-graph, in which, nodes represent simple HTML objects, texts and images, while directed edges reflect spatial relations on the browser screen.

As mentioned previously, Boer et al. (de Boer et al., 2010) has successfully classified web pages using only visual features. They classified pages in two binary variables: aesthetic value and design recency, achieving good accuracy. The authors also applied the same classification algorithm and methods to a multi-class categorization of the website topic and although the results obtained are reasonable, it was concluded that this classification is more difficult to perform.

3 CLASSIFICATION PROCESS

This section presents the work methodology used to fulfill the proposed objectives. Namely, how the process of classification of new web pages is done. In Fig. 1 it is possible to see the necessary steps to predict the class of new web pages. The algorithms were developed in C/C++ using the OpenCV library (Bradski, 2000), that runs under Windows, Linux and Mac OS X.

The next subsections present an explanation of the methods used to extract features from the images, and the construction of the respective feature descriptors. It is explained in detail the techniques used to perform feature selection.

3.1 Feature Extraction

The concept of feature in computer vision and image processing refers to a piece of information which is relevant and distinctive. For each web page, different feature descriptors (feature vector) are computed. This section describes how a descriptor of low level features which contains 166 attributes that characterize the page is obtained and how the SIFT descriptor using Bag of Words model is built.

3.1.1 Low Level Descriptor

Visual descriptors are descriptions of visual features of the content of an image. These descriptors describe elementary characteristics such as shape, color, texture, motion, among others. To built this descriptor the following features were extracted from each image: color histogram, edge histogram, tamura features and gabor features.

Color Histogram. It is a representation of the distribution of colors in an image. It can be built in any color space, but the ones used in this work is the HSV color space. It was selected because it reflects human vision quite accurately and because it mainly uses only one of its components (Hue) to describe the main properties of color in the image. The Hue histogram is constructed by discretization of the colors in the image into 32 bins. Each bin will represent an intensity spectrum. This means that a histogram provides a compact summarization of the distribution of data in an image.

Edge Histogram. An edge histogram will represent the frequency and directionality of the brightness changes in the image. The Edge Histogram Descriptor (EHD) describes the edge distribution in an image. It is a descriptor that expresses only the local edge

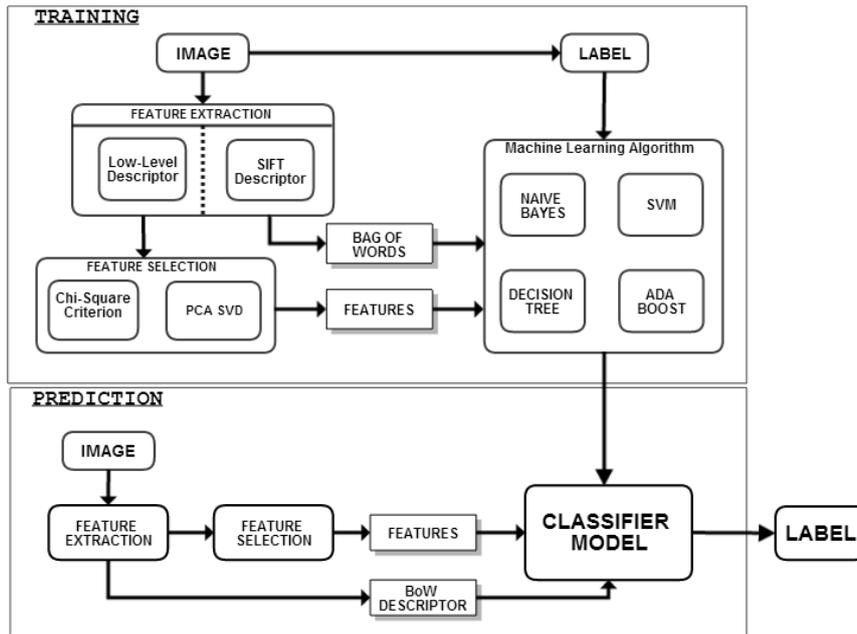


Figure 1: Classification Process diagram.

distribution in the image, describing the distribution of non-directional edges and non-edge cases, as well as four directional edges, and keeps the size of the descriptor as compact as possible for an efficient storage of the metadata. To extract the EHD, the image is divided into a fixed number of sub-images (4x4) and the local edge distribution for each sub image is represented by a histogram. The edge extraction scheme is based on an image block rather than on the pixel, i.e., each sub-image space is divided into small square blocks. For each image block it is determined which edge is predominant, i.e., the image block is classified into one of the 5 types of edge or a non edge block. Since there are 16 sub images in the image, the final histogram is constructed by $16 \times 5 = 80$ bins.

Tamura Features. Tamura et al. (Tamura et al., 1978), on the basis of psychological experiments, proposed six features corresponding to human visual perception: coarseness, contrast, directionality, line-likeness, regularity and roughness. After testing the features, the first three attained very successful results and they concluded those were the most significant features corresponding to human visual perception. The definition of these three features in (Deselaers, 2003) shows the preprocessing that is applied to the images and the steps necessary to extract those three features. The coarseness and contrast are scalar values, and the directionality is histogramized into a histogram of 16 bins.

Gabor Features. The interest about the Gabor func-

tions is that it acts as low-level oriented edge and texture discriminators, sensitive to different frequencies and scales, which motivated researchers to extensively exploit the properties of the Gabor functions. The Gabor filters have been shown to possess optimal properties in both spatial and frequency domain, and for this reason it is well suited for texture segmentation problems. Zhang et al. (Zhang et al., 2000) present an image retrieval method based on Gabor filter, where the texture features were found by computing the mean and variation of the Gabor filtered image. The final descriptor is composed by 36 attributes.

3.1.2 SIFT Descriptor using Bag of Words Model

In pattern recognition and machine learning, keypoint-based image features are getting more attention. Keypoints are salient image patches that contain rich local information of an image. The Scale Invariant Feature Transform was developed in 1999 by David Lowe. The SIFT features are one of the most popular local image features for general images, and was later refined and widely described in (Lowe, 2004). This approach transforms image data into scale-invariant coordinates relative to local features.

On the other hand, the bag-of-words (BoW) model (Liu, 2013) is a feature summarization technique that can be defined as follows. Given a training dataset D , that contains n images, where $D = \{d_1, d_2, \dots, d_n\}$, where d is the extracted features, a specific algorithm is used to group D based on a fixed number of visual

words W represented by $W = \{w_1, w_2, \dots, w_v\}$, where v is the number of clusters. Then, it is possible to summarize the data in a $n \times v$ co occurrence table of counts $N_{ij} = N(w_i, d_j)$, where $N(w_i, d_j)$ denotes how often the word w_i occurred in an image d_j .

To extract the BoW feature from images the following steps are required: i) detect the SIFT keypoints, ii) compute the local descriptors over those keypoints, iii) quantize the descriptors into words to form the visual vocabulary, and iv) to retrieve the BoW feature, find the occurrences in the image of each specific word in the vocabulary.

Using the SIFT image feature detector and descriptor implemented in OpenCV, each image is abstracted by several local keypoints. These vectors are called feature descriptors and as explained above the SIFT converts this keypoints into a 128-dimensional vector. But once we extract such local descriptors for each image, the total number of them would most likely be of overwhelming size. In that case, BoW solve this problem by quantizing descriptors into "visual words", which decreases the descriptors amount dramatically. This is done by k-means clustering, an iterative algorithm for finding clusters in data. This will allow to find a limited number of feature vectors that represent the feature space, allowing to construct the dictionary.

Once the dictionary is constructed, it is ready to be used to encode images. In the implementation of this algorithm, different sizes of the dictionary (i.e., the number of cluster centers) were used, to analyze the difference in the performance of the classifiers.

3.2 Feature Selection

An important component of both supervised and unsupervised classification problems is feature selection - a technique that selects a subset of the original attributes by selecting a number of relevant features. By choosing a better feature space, a number of problems can be solved, e.g., avoid overfitting and achieve better generalization ability, reduce the storage requirement and training time and allowing us to better understand the domain. Two algorithms for applying feature selection are built. One is based on the Chi-Square Criterion, the other uses the Principal Components Analysis. In both methods a different number R corresponding to the most relevant features is selected. The different values of R used in this work are 1%, 2%, 5%, 10%, 20% and 50% of the total features.

3.2.1 Chi-Square Criterion

Feature Selection via chi square (χ^2) test is a very commonly used method (Liu and Setiono, 1995).

Chi-squared attribute evaluates the worth of a feature by computing the value of the chi-squared statistic with respect to the class. The Feature Selection method using the Chi-Squared criterion is represented in algorithm 1.

Algorithm 1: Feature Selection using Chi-Square Criterion.

Input: Data Matrix ($M \times N$) \triangleright M represents the number of samples, and N the number of features

Input: Number of classes C .

Output: Top R features

1: For each feature and class
Find the mean value corresponding to each feature.

2: For each feature
Compute the mean value of the classes mean values.

Compute the Expected and Observed Frequencies, and calculate the chi-squared value.

$$\chi^2 = \sum \frac{(\text{ExpectedFreq} - \text{ObservedFreq})^2}{\text{ExpectedFreq}};$$

3: Sort the chi-squared values and choose the R features with the smallest sum of all values.

3.2.2 Principal Component Analysis using Singular Value Decomposition

PCA was invented in 1901 by Karl Pearson as an analogue of the principal axes theorem in mechanics. This algorithm is based on (Song et al., 2010), that proposed a method using PCA to perform feature selection. They achieved feature selection by using the PCA transform from a viewpoint of numerical analysis, allowing to select a number of M features components from all the original samples. In algorithm 2 the Singular Value Decomposition (SVD) is used to perform PCA. The SVD technique allows to reduce dimensionality by obtaining a more compact representation of the most significant elements of the data set, and this enable to express the data set more compactly.

4 WEB PAGES DATABASE

In this work, different web page classification experiments are evaluated. There are two binary classifications and one multi-category classification. The two binary classifications are: the aesthetic value of a web

Algorithm 2: Feature Selection using PCA through SVD.

Input: Data Matrix ($M \times N$) \triangleright M represents the number of samples, and N the number of features

Output: Top R features

- 1: Perform mean normalization in the Data Matrix.
 - 2: Calculate the SVD decomposition of the Data Matrix.
 - 3: Select the eigenvectors that correspond to the first d largest singular values, and denote these vectors as K_1, \dots, K_d , respectively.
 - 4: Calculate the contribution, of each feature component as follows $c_j = \sum_{p=1}^d |K_{pj}|$, where K_{pj} denotes the j entry of K_p , $j = 1, 2, \dots, N$, $p = 1, 2, \dots, d$. $|K_{pj}|$ stands for the absolute value of K_{pj} .
 - 5: Sort c_j in the descending order, and select the R features corresponding to the R largest orders in c_j .
-

page, i.e., if a web page is beautiful or ugly (a measure that depends on the notion of aesthetic of each person), and the design recency of a web page, i.e., trying to distinguish between old fashioned and new fashioned web pages. The multi category classification involves classification on the web page topic.

Using the FireShot plugin¹ for the Firefox web browser, allows to retrieve a screen shot of a web page and save it as a .PNG file. Different training sets of 30, 60 and 90 pages are built for each class of the classification experiment. For each site we only retrieved the landing page which is generally the index page.

4.1 Aesthetic

The notion of aesthetic differs from person to person, because what can be beautiful for someone, can be ugly for another. That is why this classification depends of each classifier and it is a subjective classification. Nevertheless, there is a generic notion of the beautiful and of the ugly that is common to the individuals of a certain culture. We emphasize that this underlying notion of the aesthetic value is of extremely importance to marketing and psychological explorations.

¹<https://addons.mozilla.org/pt-pt/firefox/addon/fireshot/>

In this classification experiment two classes are then defined: ugly and beautiful web pages. Notice that in Aesthetic, the important aspect is the visual design ("Look and Feel") of a web page, and not the quality of information or popularity of the page.

The ugly pages were downloaded from two articles (Andrade, 2009) and (Shuey, 2013) and their corresponding comment section, and also from the website World Worst Websites of the Year 2012 - 2005 (Flanders, 2012). The beautiful pages were retrieved, consulting a design web log, listing the author's selection of the most beautiful web pages of 2008, 2009, 2010, 2011 and 2012 (Crazyleafdesign.com, 2013).

After analyzing the web pages retrieved (Fig.2), it was possible to notice that, in general, an ugly web page don't transmit a clear message, uses too much powerful colors, lacks clarity and a consistent navigation. While, on the opposite side, it was possible to notice that a beautiful web page usually has an engaging picture, an easy navigation, the colors complement each other and it is easy to find the information needed. Obviously these are some directives observed from the database and do not correspond to strict conclusions.

4.2 Design Recency

The objective of this classification is to be able to distinguish from old fashioned and new fashioned pages. The principal differences between these pages (Fig.3) is that nowadays the web design of a page has firmly established itself as an irreplaceable component of every good marketing strategy. Recent pages usually have large background images, blended typography, colorful and flat graphics, that is, every design element brings relevant content to the user. In the past the use of GIFs, very large comprised text and blinding background were common in most sites.

The old web pages were retrieved consulting the article (waxy.org, 2010), that shows the most popular pages in 1999, and using the Internet Archive web site² allowed to retrieve the versions of those websites in that year. To retrieve the new pages, the Alexa³ web page popularity rankings was used, selecting then the 2012 most popular pages.

4.3 Web Page Topic

In this classification eight classes are defined. These classes are newspapers, hotels, celebrities, conferences, classified advertisements, social networks, gaming and video-sharing.

²<http://archive.org/web/web.php>

³<http://www.alexa.com>

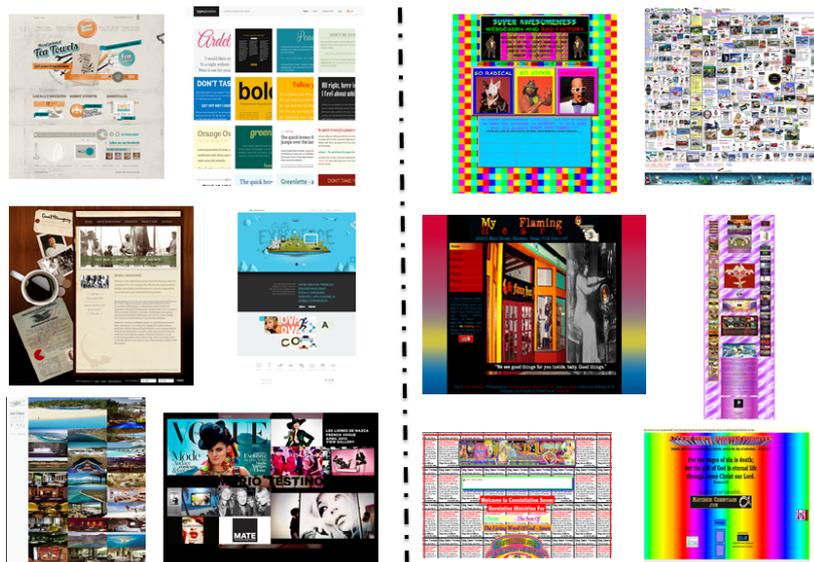


Figure 2: An example of the web pages retrieved for the Aesthetic classification. In the left, there are 6 beautiful web pages, and in the right 6 ugly web pages.

For the newspaper and celebrity classes, the Alexa.com was consulted, retrieving the most well-known and popular newspapers and celebrity sites. The celebrity sites also include popular fan sites. The conferences class consist in the homepages of the highest ranked Computer Science Conferences. And for the hotel class, different sites from bed-and-breakfast businesses are retrieved. The classes include different pages from different countries. The classified advertisements sites were extracted using also the Alexa.com, retrieving the most visited sites of classifieds of all world (sections devoted to jobs, housing, personals, for sale, items wanted, services, community, gigs and discussion forums). The video-sharing class and the gaming class (company gaming websites and popular gaming online websites), were extracted consulting the google search engine for the most popular sites in this type of websites. Social networks class consist in the major social networking websites homepages (e.g., websites that allow people to share interests, activities, backgrounds or real-life connections).

A topic of a web site is a relevant area in the classification of web pages. Each topic has a relevant visual characteristic that distinguishes them, being possible to classify the web pages despite of their language or country. Looking at the pages retrieved (Fig.4 and 5), it is possible to perceive a distinct visual characteristic in each class. The newspaper sites have a lot of text followed with images, while celebrity sites have more distinct colors and embedded videos. The conferences sites usually consist in a banner in the top of the page, and text information about the confer-

ence. Hotel sites have a more distinct background, with more photographs. Classifieds sites consist almost in blue hyperlinks with images or text, with a soft color background and banner. The body content of a video-sharing site consist in video thumbnails. The gaming sites have a distinct banner (an image or huge letters), with a color background and embedded videos. The social networks homepages, have a color pattern that is persistent.

5 RESULTS AND DISCUSSION

By training our classifiers with different training data sets, different comparisons can be made. Different evaluations were made to analyze what features and which classifiers are better for each classification task. Each classifier was evaluated with the low feature descriptor (containing 166 features), just the Color Histogram, Edge Histogram, Tamura Features, Gabor Features, and the descriptor containing the most relevant features selected by the methods of feature selection. Additionally the same data sets were used to train the classifiers with the SIFT descriptor using the bag of words model. The results for each classification task are shown in the next sections, as well as a comparison with the results of (de Boer et al., 2010). Different tests were performed using different data size for the training of the classifiers.

To test all methods after the training phase, new web pages were used to the prediction phase. Our results are based on the accuracy achieved by this prediction phase.

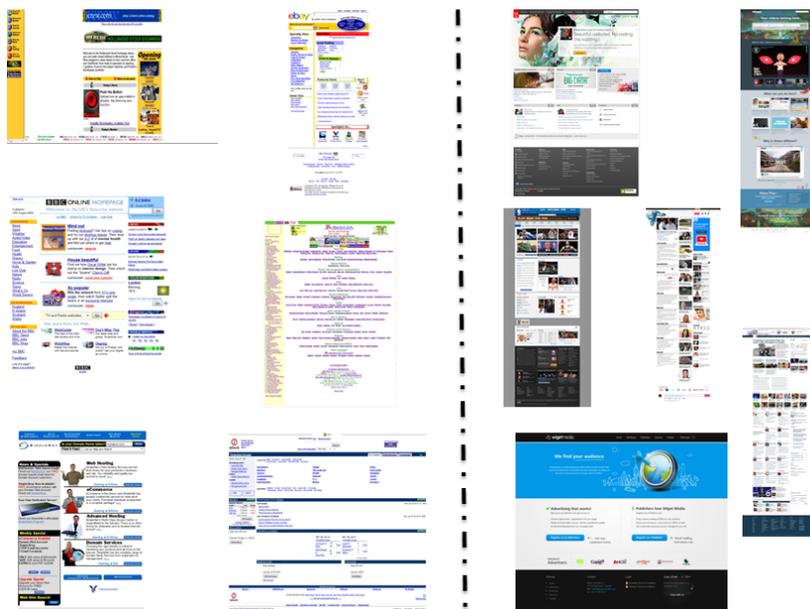


Figure 3: An example of the web pages retrieved for the Recency classification. In the left, there are 6 old fashioned web pages from 1999, and in the right 6 new fashioned web pages from 2012.



Figure 4: Examples of web pages extracted for four web site topic classes.

5.1 Aesthetic Value Results

Boer et al. (de Boer et al., 2010) in this experiment with the 166 features achieved an accuracy using the Naive Bayes and a J48 Decision Tree of 68% and 80% respectively. Using just the Simple Color Histogram and Edge Histogram they correctly classified 68% and 70% respectively for the Naive Bayes, and 66% and 53% for the J48 Decision Tree classifier.

For this experiment, Fig.6 show the best rate prediction for our classifiers, when used the SIFT de-

scriptor. Using different sizes for the dictionary, we obtained good result for each classifier. The best results for the Naive Bayes, SVM and the Decision Tree was of 80%, and for the AdaBoost we achieved a prediction accuracy of 85%.

When trained the model using just the Color Histogram attributes, the results show an accuracy of 65% for Naive Bayes, 85% in SVM, 70% for the Decision Tree and 85% using the AdaBoost when trained with 90 images for each class. When we selected the top discriminative attributes to train the classi-



Figure 5: Examples of web pages extracted for the other four web site topic classes.

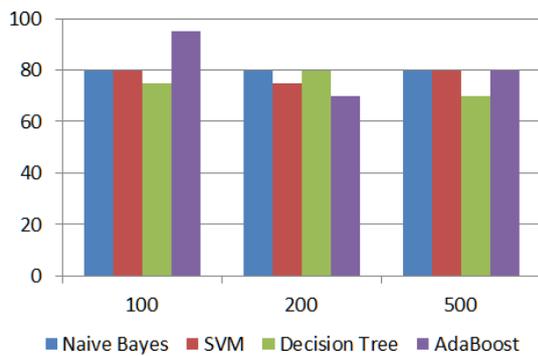


Figure 6: SIFT Descriptor using BoW Model prediction results with different dictionary sizes (100, 200 and 500) for the Aesthetic Value.

ifiers, the best results using the Chi-Squared method was when the classifiers were trained with the top 50% attributes. The Naive Bayes and SVM achieved an accuracy of 65%, the Decision Tree 80% and the AdaBoost an accuracy of 75%. When trained with the top 20% attributes by using the PCA method, the Naive Bayes classifier achieved an accuracy of 75%, the SVM classifier predicted 65% of corrected pages, and finally, the Decision Tree and the AdaBoost classifiers both had an accuracy of 80%.

All the classifiers showed a high prediction accuracy, with different features. Since most of the features chosen by the feature selection method are from the Color Histogram, it is possible to achieve a good prediction rate just by passing this simple descriptor. The SIFT descriptor give the best results, proving that the images from this two classes have distinctive key-points.

5.2 Design Recency Results

In this experiment, Boer et al. (de Boer et al., 2010) using the complete feature vector achieved an accuracy using the Naïve Bayes and a J48 Decision Tree of 82% and 85% respectively. Using just the Simple Color Histogram the Naïve Bayes performed slightly worse than the baseline and the J48 Decision Tree classifier slightly better. Using only the edge information, both models correctly classified 72% and 78% respectively for the Naïve Bayes and J48 Decision Tree classifier.

Our best results for this experiment, using the low-level descriptor, are shown in Fig.7. The Naïve Bayes, SVM and Adaboost achieved an accuracy of 100%, when the top 5% attributes were selected using the chi-square method for the first one and the Gabor descriptor for the other two. The Decision Tree best accuracy (95%), was when the PCA method selected the top 5% attributes.

Relatively to the SIFT descriptor, all the classifiers obtain a good accuracy. Noteworthy that all the classifiers obtain an accuracy of 90% when they used a dictionary size of 500. The best accuracy result achieved was for the Naïve Bayes with a 95% rate of success, with a dictionary size of 200 words.

These results proves that the classifiers can learn just by using simple visual features. All the classifiers obtained good accuracy around 85%, using just the top 1% attributes selected by both methods. Instead of using a more complex method like BoW, the use of simple visual features allows to decrease the computational cost for larger databases.

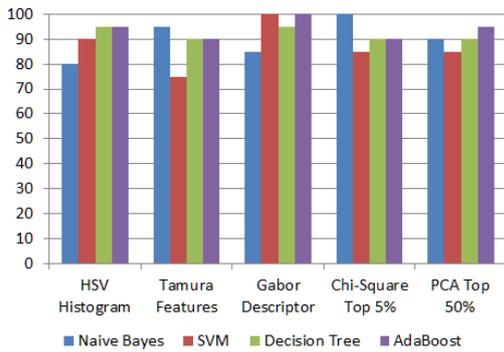


Figure 7: Best prediction results for the Recency value for four different classifiers, using the low-level descriptor. All these predictions values, were obtained by training the classifiers using 90 images for each class.

5.3 Web Page Topic Results

5.3.1 Experiment 1 - Four Classes

(de Boer et al., 2010) define the following four classes for the topic: newspapers, hotel, celebrities and conference sites. The classification results obtained were the following: when all features are used, an accuracy of 54% and 56% for the Naïve Bayes and the J48 respectively. Using the Color Histogram subset result in much worse accuracy. Using only the Edge Histogram attributes, the Naïve Bayes predict with an accuracy of 58%, whereas the J48 predicts with an accuracy of 43%. When they performed feature selection they show that the best predicting attributes are all from the Tamura and Gabor feature vectors. Using the top 10 attributes a prediction accuracy of 43% for both classifiers was obtained.

Using the same low-level descriptor that they used, all our classifiers obtained better results. The Naïve bayes achieved an accuracy of 62,5% using the Tamura Features. The SVM and Decision Tree achieved an accuracy rate of 72,5%, when used the selected top 20% attributes using the PCA method and using the whole descriptor, respectively. While the AdaBoost classifier achieved an accuracy of 70% using the PCA method selecting the top 50% attributes.

Furthermore, the results showed in Fig. 8 are an improvement of the accuracy of approximately 22% using the BoW model. Every classifier have an acceptable accuracy, where the best accuracy result is as high as 82,5% for the Decision Tree using just 100 words to construct the dictionary. In fact all the classifiers have accuracy higher than or equal to 70% when used just 100 words in the dictionary.

Table 1: Confusion Matrix for 4 classes each with 10 web pages, for the best prediction result of the Naïve Bayes classifier, using the SIFT descriptor.

| | | Actual | | | |
|-----------|--------|--------|-------|--------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel |
| Predicted | Newsp. | 7 | 0 | 0 | 0 |
| | Conf. | 2 | 7 | 2 | 2 |
| | Celeb. | 0 | 0 | 8 | 2 |
| | Hotel | 1 | 3 | 0 | 6 |



Figure 8: SIFT Descriptor using BoW Model best prediction results with different dictionary sizes (100, 200 and 500). Experiment with 4 classes.

Examining the results of the confusion matrices (Table 1, 2, 3 and 4) corresponding to the best predictions of each classifier using the SIFT with BoW model (Fig. 8), it was verified, when analyzing the accuracy by class, that the Naïve Bayes, Decision Tree and AdaBoost perform much worse for the Hotel class. The Naïve Bayes and AdaBoost classifiers reports false positives for the Hotel class as Conference or Celebrity pages. While the Decision Tree returns false positives for Celebrities web pages as Hotel web pages, and vice versa. By his hand, the SVM classifiers perform much worse for the Celebrity web pages where most of the instances are erroneously classified as Hotel pages. Since the Newspapers and Conference classes have simpler designs, when compared with the other classes, they are easier to distinguish. On the other hand, it is harder to distinguish between more complex and sophisticated classes like Hotel and Celebrity.

Although the results obtained for this multi-class categorization are worse than those obtained for aesthetic value and design recency, generally good accuracy was obtained with best values usually near or above 80%. Additionally, our results are better than those obtained by Boer et al. (de Boer et al., 2010), mainly if SIFT with BoW is used.

Table 2: Confusion Matrix for 4 classes each with 10 web pages, for the best prediction result of the **SVM** classifier, using the SIFT descriptor.

| | | Actual | | | |
|-----------|--------|--------|-------|--------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel |
| Predicted | Newsp. | 10 | 1 | 1 | 0 |
| | Conf. | 0 | 8 | 1 | 0 |
| | Celeb. | 0 | 0 | 4 | 2 |
| | Hotel | 0 | 1 | 4 | 8 |

Table 3: Confusion Matrix for 4 classes each with 10 web pages, for the best prediction result of the **Decision Tree** classifier, using the SIFT descriptor.

| | | Actual | | | |
|-----------|--------|--------|-------|--------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel |
| Predicted | Newsp. | 10 | 0 | 1 | 0 |
| | Conf. | 0 | 9 | 0 | 1 |
| | Celeb. | 0 | 1 | 7 | 2 |
| | Hotel | 0 | 0 | 2 | 7 |

Table 4: Confusion Matrix for 4 classes each with 10 web pages, for the best prediction result of the **AdaBoost** classifier, using the SIFT descriptor.

| | | Actual | | | |
|-----------|--------|--------|-------|--------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel |
| Predicted | Newsp. | 10 | 0 | 1 | 0 |
| | Conf. | 0 | 6 | 1 | 3 |
| | Celeb. | 0 | 1 | 8 | 3 |
| | Hotel | 0 | 2 | 0 | 4 |

5.3.2 Experiment 2 - Eight Classes

Along with the four classes defined in the experiment 1, four additional classes were added to this classification: classified advertisements sites, gaming sites, social networks sites and video-sharing sites.

Using the low-level descriptor the Naïve Bayes had the best accuracy with 47,5%, while the SVM achieved an accuracy of 41,25% using the Tamura descriptor. The Decision Tree and AdaBoost classifiers had a poor performance, where the best accuracy was 37,5% and 33,75%, respectively. When we used the Chi-Squared and PCA method to select the top attributes the classifiers performance didn't improve. We conclude that for this type of classification more complex features or a bigger database are necessary.

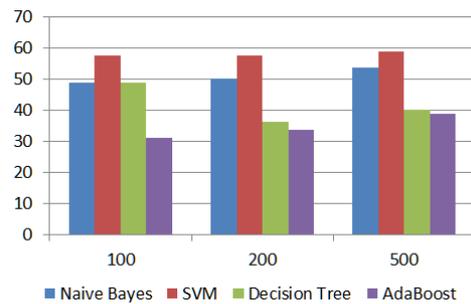


Figure 9: SIFT Descriptor using BoW Model best prediction results with different dictionary sizes (100, 200 and 500). All these predictions values, were obtained by training the classifiers using 30 and 60 images for each class.

When we used the SIFT descriptor (Fig. 9) all the classifiers had a better accuracy relatively to the results obtained using the low-level descriptor. The SVM achieved an accuracy of 58,75%, and the Naïve Bayes 63,75%. The Decision Tree best accuracy was 48,75% , while the Adaboost only predict the correct class in 38,75% of the predictions.

When examining the confusion matrices (Table 5 and 6) of Naïve Bayes and SVM classifiers (which achieved accuracy over 50% when using the SIFT descriptor), it is possible to verify that both classifiers have problems distinguishing celebrities web pages. The Naïve Bayes also struggles in identify Video-Sharing pages (only 3 correct predictions), while the SVM have troubles in identifying Social Networks web pages (only 2 correct predictions). The body of video-sharing web pages that consist mostly in video thumbnails are easily mistaken as newspapers web page (mostly images followed by text). In both methods some classified advertisements web pages are also predicted as newspapers (most classified advertisement websites use a simple color background with a lot of images). To overcome this drawbacks a bigger database is necessary.

5.4 Discussion

The results show that based on aesthetic value and design recency, simple features such as color histogram and edges provide quite good results, where in some cases an accuracy of 100% is achieved (average best accuracy of 85%). For the topic classification, the use of a SIFT with BoW provide much better results.

As expected when more website topics are added to topic classification, the classification gets harder and the classifiers accuracy decreases to an average accuracy of around 60%. This indicates that even if the pages have visual characteristics that distinguishes them, they also have some attributes or characteris-

Table 5: Confusion Matrix for 8 classes, for the best prediction result of the **Naïve Bayes** classifier, using the SIFT descriptor.

| | | Actual | | | | | | | |
|-----------|-----------|--------|-------|--------|-------|----------|--------|-----------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel | Classif. | Gaming | Social N. | Video |
| Predicted | Newsp. | 9 | 0 | 1 | 1 | 3 | 1 | 0 | 4 |
| | Conf. | 1 | 5 | 0 | 0 | 1 | 0 | 0 | 0 |
| | Celeb. | 0 | 0 | 3 | 2 | 0 | 2 | 2 | 1 |
| | Hotel | 0 | 1 | 0 | 5 | 0 | 1 | 1 | 0 |
| | Classif. | 0 | 1 | 1 | 1 | 6 | 0 | 0 | 1 |
| | Gaming | 0 | 0 | 5 | 0 | 0 | 6 | 0 | 0 |
| | Social N. | 0 | 1 | 0 | 1 | 0 | 0 | 6 | 1 |
| | Video | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 3 |

Table 6: Confusion Matrix for 8 classes, for the best prediction result of the **SVM** classifier, using the SIFT descriptor.

| | | Actual | | | | | | | |
|-----------|-----------|--------|-------|--------|-------|----------|--------|-----------|-------|
| | | Newsp. | Conf. | Celeb. | Hotel | Classif. | Gaming | Social N. | Video |
| Predicted | Newsp. | 9 | 1 | 1 | 1 | 4 | 0 | 1 | 2 |
| | Conf. | 1 | 8 | 0 | 0 | 0 | 0 | 0 | 0 |
| | Celeb. | 0 | 0 | 4 | 2 | 0 | 3 | 2 | 1 |
| | Hotel | 0 | 0 | 0 | 7 | 0 | 1 | 1 | 1 |
| | Classif. | 0 | 0 | 1 | 0 | 6 | 1 | 0 | 0 |
| | Gaming | 0 | 0 | 4 | 0 | 0 | 5 | 2 | 0 |
| | Social N. | 0 | 1 | 0 | 0 | 0 | 0 | 2 | 0 |
| | Video | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 6 |

tics in common. To overcome this setbacks a bigger database is necessary. Nevertheless, the aim of this work was to demonstrate that it is possible to classify web pages in different topics with reasonable accuracy and to prove that this visual content is very rich and can be successfully used to complement, not to substitute, the current classification by crawlers that use only text information. Notice too, that in the design of web pages, there is a growing tendency to include content in the images used, preventing text-based crawlers to get to this rich content (mainly in titles, separators and banners).

Classification using the visual features has however some limitations: if the image of the web page has poor quality, the accuracy in the classification will drastically be reduced. Other disadvantage is that many web page topics have very common patterns in their design, making very hard to the classifier to distinguish between them. We intend to enhance these

classifiers in the future to improve its accuracy.

6 CONCLUSION

In this work we described an approach for the automatic web page classification by exploring the visual content "Look and feel" of web pages, as they are rendered by the web browser. The results obtained are quite encouraging, proving that the visual content of a web page should not be ignored, when performing classification. This implementation uses a method for categorization based on low-level features.

In the future, in order to improve the classification accuracy we can also follow some additional paths. The integration of these visual features with other features of web pages can thus boost the accuracy in the classifiers. The analysis of the visual appearance of a web page can be combined with the well-established

analysis based on text content, URL, the underlying HTML, or others. In this case associate this visual features with the text content may give rise to a powerful classification system. Additionally, we also intend to mix the classification using visual features with a semantic analysis of them. We expect to improve the results by integrating the semantic content of a webpage image not only in the classification of the aesthetic or recency value but also for the classification of the topic. Another approach is the extraction of more sophisticated features that can analyze their dynamic elements (animated gifs, flash, advertisement content, and so on).

As for the applications of the visual classification of web pages, the methods studied may be applied to an advice system that assist the design and rating of web sites that can be applied to content filtering. In a research perspective, the fact that the aesthetic and design recency value are such a subjective measures, also make of great interest studies of the consumer profile for the field of digital marketing.

ACKNOWLEDGEMENTS

The authors acknowledge the support of the Portuguese Science Foundation through project PEst-C/EEI/UI0048/2013.

REFERENCES

- Andrade, L. (2009). The worlds ugliest websites!!! retrieved october 2009: <http://www.nikibrown.com/designoblog/2009/03/03/the-worlds-ugliest-websites/>.
- Asirvatham, A. P. and Ravi, K. K. (2001). Web page classification based on document structure. In *IEEE National Convention*.
- Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
- Chen, R. C. and Hsieh, C. H. (2006). Web page classification based on a support vector machine using a weighted vote schema. *Expert Syst. Appl.*, 31(2):427–435.
- Crazyleafdesign.com (2013). Most beautiful and inspirational website designs.
- de Boer, V., van Someren, M., and Lupascu, T. (2010). Classifying web pages with visual features. In *WEBIST (2010)*, pages 245–252.
- Deselaers, T. (2003). Features for image retrieval (thesis). Master's thesis, RWTH Aachen University, Aachen, Germany.
- Flanders, V. (2012). Worst websites of the year 2012 - 2005: <http://www.webpagesthatsuck.com/worst-websites-of-the-year.html>.
- Kovacevic¹, M., Diligenti, M., Gori, M., and Milutinovic¹, V. (2004). Visual adjacency multigraphs, a novel approach for a web page classification. *Workshop on Statistical Approaches to Web Mining (SAWM)*, pages 38–49.
- Liu, H. and Setiono, R. (1995). Chi2: Feature selection and discretization of numeric attributes. In *Proceedings of the Seventh International Conference on Tools with Artificial Intelligence*, TAI '95.
- Liu, J. (2013). Image retrieval based on bag-of-words model. *arXiv preprint arXiv:1304.5168*.
- Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110.
- Selamat, A. and Omatu, S. (2004). Web page feature selection and classification using neural networks. *Inf. Sci. Inf. Comput. Sci.*, pages 69–88.
- Shuey, M. (2013). 10-worst-websites-for-2013: <http://www.globalwebfx.com/10-worst-websites-for-2013/>.
- Song, F., Guo, Z., and Mei, D. (2010). Feature selection using principal component analysis. In *System Science, Engineering Design and Manufacturing Informatization (ICSEM), 2010 International Conference on*, volume 1, pages 27–30.
- Tamura, H., Mori, S., and Yamawaki, T. (1978). Textural features corresponding to visual perception. *IEEE Transaction on Systems, Man, and Cybernetics*, 8:460–472.
- waxy.org (2010). Den.net and the top 100 websites of 1999: http://waxy.org/2010/02/dennet_and_the_top_100_websites_of_1999/.
- Zhang, D., Wong, A., Indrawan, M., and Lu, G. (2000). Content-based image retrieval using gabor texture features. In *IEEE Pacific-Rim Conference on Multimedia, University of Sydney, Australia*.

User Semantic Model for Dependent Attributes to Enhance Collaborative Filtering

Sonia Ben Ticha^{1,2}, Azim Roussanaly¹, Anne Boyer¹ and Khaled Bsaïes²

¹ LORIA-KIWI Team, Lorraine University, Nancy, France

² LIPA Lab, Tunis El Manar University, Tunis, Tunisia

{sonia.benticha, azim.roussanaly, anne.boyer}@loria.fr, khaled.bsaies@fst.rnu.tn

Keywords: Hybrid Recommender System, Latent Semantic Analysis, Rocchio Algorithm.

Abstract: Recommender system provides relevant items to users from huge catalogue. Collaborative filtering and content-based filtering are the most widely used techniques in personalized recommender systems. Collaborative filtering uses only the user-ratings data to make predictions, while content-based filtering relies on semantic information of items for recommendation. Hybrid recommendation system combines the two techniques. The aim of this work is to introduce a new approach for semantically enhanced collaborative filtering. Many works have addressed this problem by proposing hybrid solutions. In this paper, we present another hybridization technique that predicts users preferences for items based on their inferred preferences for semantic information of items. For this, we design a new user semantic model by using Rocchio algorithm and we apply a latent semantic analysis to reduce the dimension of data. Applying our approach to real data, the MoviesLens 1M dataset, significant improvement can be noticed compared to usage only approach, and hybrid algorithm.

1 INTRODUCTION

Recommender Systems (RS) provide relevant items to users from a large number of choices. Several recommendations techniques exist in the literature. Among these techniques, there are those that provide personalized recommendations by defining a profile for each user. In this work, we are interested in personalized recommender systems where the user model is based on an analysis of usage. This model is usually described by a user-item ratings matrix, which is extremely sparse ($\geq 90\%$ of missing data).

Collaborative Filtering (CF) and Content-Based (CB) filtering are the most widely used techniques in RS. The fundamental assumption of CF is that if users X and Y rate n items similarly and hence will rate or act on other items similarly (Su and Khoshgof-taar, 2009). CB filtering assumes that each user operates independently and user will be recommended items similar to the ones he preferred in the past (Lops et al., 2011). The major difference between CF and CB recommender systems is that CF uses only the user-item ratings data to make predictions and recommendations, while CB relies on item content (semantic information) for recommendations. However, CF and CB techniques must face many challenges like

the data sparsity problem, the scalability problem for large data with the increasing numbers of users and items.

To overcome the disadvantages of both techniques and benefit from their strengths, hybrid solutions have emerged. In this paper, we present a new approach taking into account the semantic information of items in a CF process. In our approach, we design a new hybridization technique, which predicts user preferences for items based on their inferred preferences for latent item content; and presents a solution to the sparsity and scalability problems. Our system consists of two components: the first builds a new user model, *the user semantic model*, by inferring user preferences for item content; the second computes predictions and provides recommendations by using the user semantic model in a user-based CF algorithm (Resnick et al., 1994) to calculate the similarity between users. The originality of this work is in the building of the user semantic model. Indeed, assuming that items are represented by structured data in which each item is described by a same set of attributes, we build a *user semantic attribute model* for each relevant attribute. With this aim, we define two classes of attributes: *dependent* and *non dependent* and we propose a suited algorithm for each class. User semantic model is

then deducted from the horizontal concatenation of all user semantic attribute model. In previous works (Ben Ticha et al., 2012; Ben Ticha et al., 2011) we have presented solutions based on machine learning algorithm to build a user semantic attribute model for non dependent attribute. In this work, we present a new approach for building a user semantic attribute model for dependent attribute by using Rocchio algorithm (Rocchio, 1971). Due to the high number of attribute values, and to reduce the expensiveness of user similarity computing, we apply a Latent Semantic Analysis (LSA)(Dumais, 2004) to reduce the size of the user semantic attribute model. We compare our results to the standards user-based CF, item-based CF and hybrid algorithms. Our approach results in an overall improvement in prediction accuracy.

The rest of paper is organized as follows: Section 2 summarizes the related work. User semantic model is described in Section 3. Section 4 describes our approach to build user semantic attribute model for non dependent attribute. Section 5 describes the recommendation component of our system. Experimental results are presented and discussed in Section 6. Finally, we conclude with a summary of our findings and some directions for future work.

2 RELATED WORK

Recommender System (RS) have become an independent research area in the middle 1990s. CF is the most widespread used technique in RS, it was the subject of several researches (Resnick et al., 1994; Sarwar et al., 2001). In CF, user will be recommended items that people with similar tastes and preferences liked in the past (Adomavicius and Tuzhilin, 2005). CB is another important technique; it uses techniques developed in information filtering research (Pazzani and Billsus, 2007). CB assumes that each user operates independently and recommends items similar to the ones he preferred in the past. Hybrid approach consists on combining CF and CB techniques. The Fab System (Balabanovic and Shoham, 1997) counts among the first hybrid RS. Many systems have been developed since (Burke, 2007). Most of these hybrid systems do not distinguish between attributes and treat their values in a same way. Moreover, because of the huge number of items and users, calculating the similarity between users in CF algorithm became very expensive in time computing. Dimension reduction of data is one of the solution to reduce the expensiveness of users similarity computing. Mobasher et al. (Mobasher et al., 2003) combine values of all attributes and then apply a LSA to

reduce dimension of data. Sen et al. (Sen et al., 2009) are inferring user preferences for only one attribute, the item' tags, without reducing dimension. Manzato (Manzato, 2012) computes a user semantic model for only the movie *genre* attribute and applies a Singular Value Decomposition (SVD) to reduce the dimension of data. In our approach, we compute a user semantic attribute model for each relevant attribute and we apply a suited reduction dimension algorithm for each attribute class.

3 USER SEMANTIC MODEL

In this paper, we are interested only to items described by structured data. According to the definition of Pazzani et al. (Pazzani and Billsus, 2007), in structured representation, item can be represented by a small number of attributes, and there is a known set of values that each attribute may have, for instance, the attributes of a movie can be *title*, *genre*, *actor* and *director*. In the following, we will use the term *feature* to refer to an attribute value, for instance *Documentary*, *Musical* and *Thriller* are features of *movie genre* attribute.

3.1 Dependent and Non Dependent Attribute

In structured representation, each attribute has a set of restricted features. However, the number of features can be related or not to the number of items. That is why we have defined two classes of attributes:

- **Dependent Attribute:** attribute, which having very variable number of features. This number is closely related to the number of items. So, when the number of items is increasing, the number of features is increasing also. For example: *directors* and *actors of movies*, *user tags*.
- **Non Dependent Attribute:** attribute, which having a very few variable number of features, and this number is not related to the number of items. Thus, the increasing number of items has no effect on the number of features. For example: *movie genre*, *movie origin* and *cuisine of restaurants*.

In addition, all attributes do not have the same degrees of importance to users. There are attributes more relevant than others. For instance, the *movie genre* can be more significant, in the evaluation criteria of user, than the *origin*. Experiments that we have conducted (see Section 6.2) confirmed this hypothesis. In this paper, we assume that relevant attributes will be provided by a human expert. Therefore, for

each relevant attribute A , we build a *user semantic attribute model* that predicts the users preferences for its features (or group of features). This model is described by a matrix Q_A (users in lines and features (or group of features) of A in columns). In our approach, we design a suited algorithm for building the *user semantic attribute model* for each class of attribute. For non dependent attribute, due to the low number of features, we have used a clustering algorithm. Section 3.2 briefly described the operating principle of our solution that have been addressed in previous works (Ben Ticha et al., 2012; Ben Ticha et al., 2011). For dependent attribute, we have explored techniques issues from information retrieval (IR) research. Section 4 presents our solution for building the *user semantic attribute model* for dependent attribute that is the aim of this paper. The user semantic model for all relevant attributes, described by the matrix Q , is the result of the horizontal concatenation of all user semantic attribute models Q_A .

3.2 User Semantic Model for Non Dependent Attribute

Let us denote by S the set of items, U the set of users, s a given item $\in S$, u a given user $\in U$ and a rating value $r \in \{1, 2, \dots, 5\} \equiv R$. U_s the set of users that rating the item s , then we define the rating function for item s by $\delta_s : u \in U_s \mapsto \delta_s(u) \in R$. We denote also by F_A the set of features of attribute A , f a given feature $\in F_A$ and S_f the set of items associated to feature f . For instance if we consider the *movie genre* attribute, S_{action} is the set of all action movies.

An item s is represented by its usage profile vector $s_{up} = (\delta_s(u) - \bar{\delta}_u)_{(u=1..|U|)}$, where $\bar{\delta}_u$ is the average rating of all rated items by user u . The idea is to partition all items described by their usage profile in K clusters, each cluster is labeled by a feature $f \in F_A$ (or a set of features).

The number K of clusters and the initial center of each cluster is computed by the initialization step of the clustering algorithm. In initial step, each cluster C_k consists of items in $\bigcup_{f \text{ labeling } C_k} S_f$ and labeled by the set of corresponding features; so its center is the mean of its items described by their usage profile vector s_{up} . Moreover, an attribute can be mono valued or multivalued depending on the number of features that can be associated to a given item s . For example, the attribute *movie genre* is multivalued because a movie can have several genres while *movie origin* is a mono valued attribute because a movie has only one origin. Thus, if an attribute is multivalued, s can belong to several clusters C_k , while for mono valued attribute, an item should belong only to one

cluster. Therefore, for multivalued attribute, the clustering algorithm should provide non disjointed clusters (a fuzzy clustering), whereas, for mono valued attribute, the clustering algorithm should provide disjointed clusters.

After running the clustering algorithm, we obtain K cluster centers; each center k is described by a vector $c_k = (q_{k,u})_{(u=1..|U|)}$. The K centers is modeling k latent variables issued from the features of the attribute A . Thus, the user semantic attribute model is described by the matrix $Q_A = (q_{u,k})_{(u=1..|U|, k=1..K)}$.

With non dependent attribute, the number of associated features is low, this is why the clustering is suitable. Moreover, the user semantic attribute model allows an important reduction of dimension and so reduce the expensiveness of user similarity computing. In (Ben Ticha et al., 2011), we have used the Fuzzy CMean Algorithm on the movie *genre* attribute, we have obtained good performance because the user semantic attribute model has no missing values and all similarities between users were able to be computed. In (Ben Ticha et al., 2012), we have used the KMean clustering algorithm on the movie *origin* attribute. Because of the missing values in the user item rating matrix, we have proposed an algorithm for the initialization step of the KMean clustering using a movie origin ontology. We obtained good results compared to user-based CF but not as good as results for the *genre* attribute.

4 USER SEMANTIC MODEL FOR DEPENDENTS ATTRIBUTES

For a dependent attribute A , the set F_A of its features can be important and it augments with the increasing of the set of items S . In this paper, we present our solution to compute a user semantic attribute model for dependent attribute.

In addition to the formalism used in Section 3.2, we denote by F_{A_s} the set of features $f \in F_A$ associated to item s and by S_u the set of items $s \in S$ rated by user u . We define also, the rating function of user u as $\delta_u : s \in S_u \mapsto \delta_u(s) \in R$; and the Item Frequency Function for item $s \in S$ as $freq_s : f \in F_A \mapsto 1$ if $f \in F_{A_s}$ (f associated to item s), 0 otherwise. The Frequency Item Matrix $F = (freq_s(f))_{s \in S \text{ and } f \in F_A}$ is provided by computing $freq_s(f)$ for all items and all features.

4.1 Computing the TF-IDF on the Frequency Item Matrix F

One of the best-known measures for specifying keyword weights in Information Retrieval is the TF-IDF (Term Frequency/Inverse Document Frequency) (Salton, 1989). It is a numerical statistic, which reflects how important a word is to a document in a corpus. In our case, we replace document by item and term by feature and compute TF-IDF on the Frequency Item Matrix F .

$$FF(f, s) = \frac{freq_s(f)}{\max_j freq_s(j)} \quad (1)$$

where the maximum is computed over the $freq_s(j)$ of all features in F_{A_s} of item s .

The measure of Inverse Document Frequency (IDF) is usually defined as:

$$IDF(f) = \log \frac{|S|}{|S_f|} \quad (2)$$

where $|S_f|$ is the number of items assigned to feature f (ie $freq_s(f) \neq 0$). Thus, the FF-IDF weight of feature f for item s is defined as:

$$\omega(s, f) = FF(f, s) \times IUF(f) \quad (3)$$

4.2 Rocchio Formula for User Semantic Attribute Model

Rocchio algorithm (Rocchio, 1971) is a relevance feedback procedure, which is used in information retrieval. It designed to produce improved query formulations following an initial retrieval operation. In a vector processing environment both the stored information document D and the requests for information R can be represented as t -dimensional vectors of the form $D = (d_1, d_2, \dots, d_t)$ and $B = (b_1, b_2, \dots, b_t)$. In each case, d_i and b_i represent the weight of term i in D and B , respectively. A typical query-document similarity measure can then be computed as the inner product between corresponding vectors.

Rocchio showed in (Rocchio, 1971), that in a retrieval environment that uses inner product computations to assess the similarity between query and document vectors, the best query leading to the retrieval of many relevant items from a collection of documents is:

$$B_{opt} = \frac{1}{|R|} \sum_R \frac{D_i}{|D_i|} - \frac{1}{|NR|} \sum_{NR} \frac{D_i}{|D_i|} \quad (4)$$

Where D_i represent document vectors, and $|D_i|$ is the corresponding Euclidean vector length; R is the set of relevant documents and NR is the set of non relevant documents.

We use the Rocchio formula (4) for computing the user semantic profile of user u . In our case we replace D by S the collection of items and term by feature. Thus, the user semantic model $Q_A(u)$ for user u and attribute A is equal to Q_{opt} in formula (4). The set of relevant items R for user u is composed of all items in S having $\delta_u(s) \geq \bar{\delta}_u$. The set of non relevant items NR for user u is composed of all items in S having $\delta_u(s) < \bar{\delta}_u$.

4.3 Latent Semantic Analysis for Dimension Reduction

Latent Semantic Analysis (LSA) (Dumais, 2004) is a dimensionality reduction technique which is widely used in information retrieval. Given a term-document frequency matrix, LSA is used to decompose it into two matrices of reduced dimensions and a diagonal matrix of singular values. Each dimension in the reduced space is a latent factor representing groups of highly correlated index terms. Here, we apply this technique to create a reduced dimension space for the user semantic attribute model. In fact, for dependent attribute, the number of feature is correlated to the number of items, and so it can be very elevated and even higher than the number of items. Thus, the semantic user attribute model can have dimension greater than the user rating matrix thereby aggravating the scalability problem.

Singular Value Decomposition (SVD) is a well known technique used in LSA to perform matrix decomposition. In our case, we perform SVD on the frequency item matrix $F_{|S| \times |F_A|}$ by decomposing it into three matrices:

$$F = I_{|S|,r} * \Sigma_{r,r} * V_{r,|F_A|}^T \quad (5)$$

where I and V are two orthogonal matrices; r is the rank of matrix F , and Σ is a diagonal matrix, where its diagonal entries contain all singular values of matrix F and are stored in decreasing order. I and V matrices are the left and right singular vectors, corresponding to item and feature vectors in our case. LSA uses a truncated SVD, keeping only the k largest singular values and their associated vectors, so

$$F' = I_k * \Sigma_k * V_k^T \quad (6)$$

F' is the rank- k approximation to F , and is what LSA uses for its semantic space. The rows in I_k are the item vectors in LSA space and the rows in V , are the feature vectors in LSA space. In the resulting Frequency Item Matrix, F' , each item is, thus, represented by a set of k latent variables, instead of the original features. This results in a much less sparse matrix, improving the results of users similarity computations

in CF process. Furthermore, the generated latent variables represent groups of highly correlated features in the original data, thus potentially reducing the amount of noise associated with the semantic information.

In summary, for building the user semantic attribute matrix Q_A for a dependent attribute A ; first, we apply a TF-IDF measure on Frequency Item Matrix F ; second, we reduce the dimension of Frequency Item Matrix F by applying a LSA; third, we compute the user semantic attribute model by using Rocchio formula (4).

5 RECOMMENDATION

To compute predictions for the active user u_a , we use the user-based CF algorithm (Resnick et al., 1994). User-Based CF predicts the rating value of active user u_a on non rated item $s \in S$, it is based on the k-Nearest-Neighbors algorithm. A subset of nearest neighbors of u_a are chosen based on their similarity with him or her, and a weighted aggregate of their ratings is used to generate predictions for u_a . Equation 7 provides formula for computing predictions.

$$p(u_a, s) = \bar{\delta}_{u_a} + L \sum_{v \in V} sim(u_a, v) (\delta_v(s) - \bar{\delta}_v) \quad (7)$$

where $L = \frac{1}{\sum_{v \in V} |sim(u_a, v)|}$ and V is the set of the nearest neighbors (most similar users) to u_a that have rated item s . V can range anywhere from 1 to the number of all users.

$$sim(u, v) = \frac{\sum_k (q_{u,k} - \bar{q}_u)(q_{v,k} - \bar{q}_v)}{\sqrt{\sum_k (q_{u,k} - \bar{q}_u)^2} \sqrt{\sum_k (q_{v,k} - \bar{q}_v)^2}} \quad (8)$$

The function $sim(u, v)$ provides the similarity between users u and v and is computed by using the Pearson Correlation (8). In the standard user-based CF algorithm, the users-items rating matrix $(\delta_u(s)_{(u \in U, s \in S)})$ is used to compute users' similarities. In our algorithm, for computing the similarities between users we use instead the user semantic matrix Q . As we have already mentioned, the matrix Q is the horizontal concatenation of user semantic attribute model Q_A for each relevant attribute A .

Although we apply a user-based CF for recommendation, our approach is also a model-based method because it is based on a new user model to provide ratings of active user on non rated items. Our approach resolves the scalability problem for several reasons. First, the building process of user semantic model is fully parallelizable (because the computing of user semantic attribute model is done in independent way for each other) and can be done off

line. Second, this model allows a dimension reduction since the number of columns in the user semantic model is much lower than those of user item rating matrix, so, the computing of similarities between users is less expensive than in the standard user-based CF. In addition, our approach allows inferring similarity between two users even when they have any co-rated items because the users-semantic matrix has less missing values than user item ratings matrix. Thus, our approach provides solution to the neighbor transitivity problem emanates from the sparse nature of the underlying data sets. In this problem, users with similar preferences may not be identified as such if they haven't any items rated in common.

6 PERFORMANCE STUDY

In this section, we study the performance of our approach, User Semantic Collaborative Filtering (USCF in plots), against the standards CF algorithms: User-Based CF(UBCF) (Resnick et al., 1994), and Item-Based CF(IBCF) (Sarwar et al., 2001) and an hybrid algorithm. We evaluate these algorithms in terms of predictions accuracy by using the Mean Absolute Error (MAE) (Herlocker et al., 2004), which is the most widely used metric in CF research literature. It computes the average of the absolute difference between the predictions and true ratings in the test data set, lower the MAE is, better is the prediction.

We have experimented our approach on real data from the MovieLens1M dataset of the MovieLens recommender system¹. The MovieLens1M provides the usage data set and contains 1,000,209 explicit ratings of approximately 3,900 movies made by 6,040 users. For the semantic information of items, we use the HetRec 2011 dataset (HetRec2011, 2011) that links the movies of MovieLens dataset with their corresponding web pages at Internet Movie Database (IMDb) and Rotten Tomatoes movie review systems. We use *movie genre* and *movie origin* as non dependent attributes, *movie director* and *movie actor* as dependent attributes.

We have filtered the data by maintaining only users with at least 20 ratings, and available features for all movies. After the filtering process, we obtain a data set with 6020 users, 3552 movies, 19 genres, 44 origins, 1825 directors and 4237 actors. The usage data set has been sorted by the timestamps, in ascending order, and has been divided into a training set (including the first 80% of all ratings) and a test set (the last 20% of all ratings). Thus, ratings of each user in

¹<http://www.movielens.org>

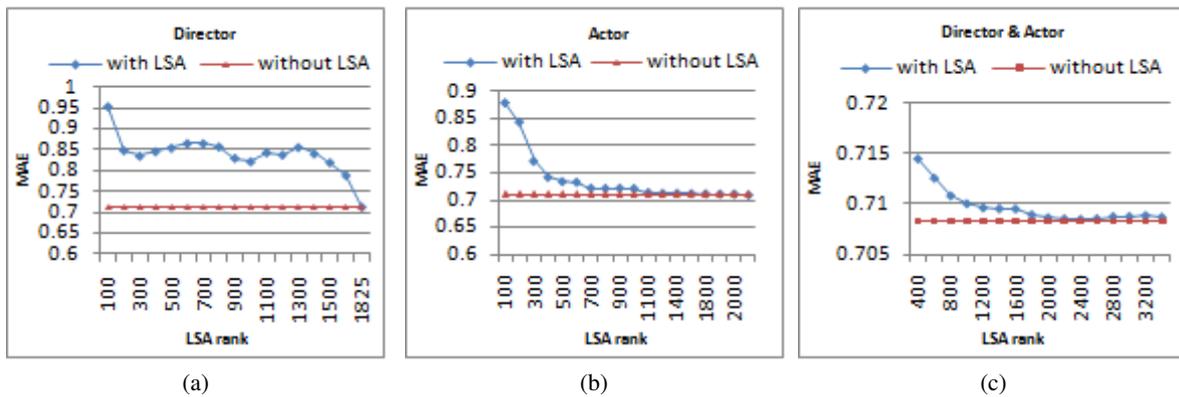


Figure 1: Impact of LSA on prediction accuracy of Rocchio algorithm.

test set have been assigned after those of training set. It should be noted that the building of user semantic attribute model for the non dependent attributes *genre* and *origin* have been addressed respectively in previous works (Ben Ticha et al., 2011; Ben Ticha et al., 2012). Therefore, we will not detail the experiments conducted for these attributes in this paper. If it is not specified, the number of nearest neighbors is equal to 60.

6.1 Impact of LSA on Prediction Accuracy

In Figure 1, the MAE has been plotted with respect to the LSA rank. It compares the Rocchio approach *with* and *without* applying LSA (dimension reduction) on *director* attribute (Figure 1(a)), *actor* attribute (Figure 1(b)) and combined attribute *director_actor* (Figure 1(c)). In all cases, the plots have the same look, the MAE of Rocchio with LSA decreases until it reaches the MAE value of Rocchio without LSA. So, LSA dimension reduction has no effect on improving the accuracy. This can be explained by the fact that features are not highly correlated, which is understandable especially for attributes *director* and *actor*, hence their poor performance. Indeed, for the *director* attribute, for instance, the MAE without reduction (1825 features) is equal to 0.7122 while the best value with LSA is equal to 0.7884. However, for combined attributes *director_actor* (6062 features), the best value is equal to 0.7083 (obtained for Rocchio without LSA) while the worst value is equal to 0.7145 (Rocchio with LSA, rank=400). For rank equal to 1200, MAE= 0.7096, so a dimension reduction about 80% for a loss of accuracy about 0.18%. In this case, features of combined attribute, *actor_director*, are more correlated than the features of each attribute taken alone hence, the best performance. Although the LSA doesn't improve the accuracy, dimension re-

duction is significant. Thus, it allows to reduce the cost of users similarity computing, specially when the number of features is very high, as is the case of combined attributes *director_actor*.

6.2 Impact of Attribute Class on Prediction Accuracy

Figure 2 compares algorithms for building user semantic attribute model in term of MAE. The *Average* algorithm (Average in plot) is building user semantic attribute model by computing the average of user ratings by feature ($q_{(u,f)} = AVG\{\delta_u(s)/s \in S_u \text{ and } f \in F_{A_s}\}$). *Fuzzy C Mean* algorithm (FuzzyCM in plot) is a fuzzy clustering used for non dependent and multivalued attribute (here *genre*) and *KMean* algorithm (KMean in plot) is used on non dependent and mono valued attribute (here *origin*). Moreover, Rocchio algorithm (Rocchio in plot) is applied here for all attributes dependent and non dependent. For *genre*, *origin* and *director* attributes, Rocchio without LSA provides best results than with dimension reduction. For *actor* attribute, LSA with rank equal to 1100 is applied (Rocchio+LSA in plot). When analyzing this figure we note first, that *Average* algorithm provides, for all attributes, the worst performance compared to all other algorithms. Second, if we applied the *Rocchio* algorithm to non dependent attribute the performance compares unfavorably against the dependent attribute, while the best performance is attained by *FuzzyCM* algorithm on *genre* attribute and the difference is important (0.7079 for FuzzyCM and 0.7274 for Rocchio). This allows to deduct that, using a suited algorithm for each attribute class provides best performance than applying the same algorithm for all attributes. Third, the *origin* attribute has the worst performance compared to the other three attributes and this for all algorithms; this is confirm our hypothe-

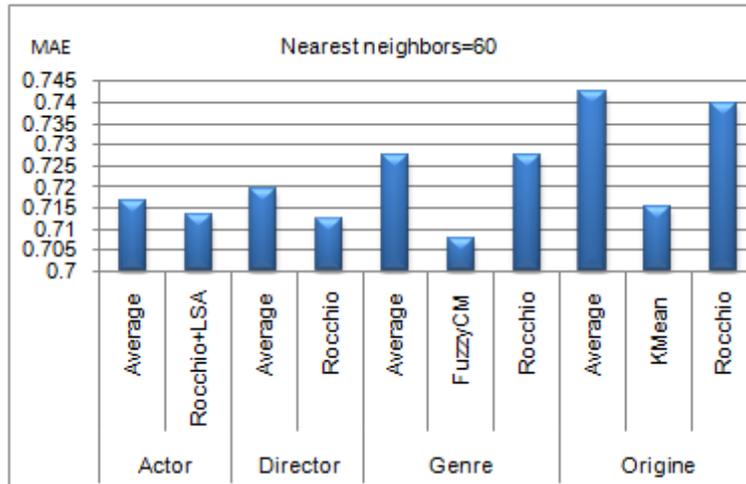


Figure 2: Impact of user semantic attribute algorithm on prediction accuracy.

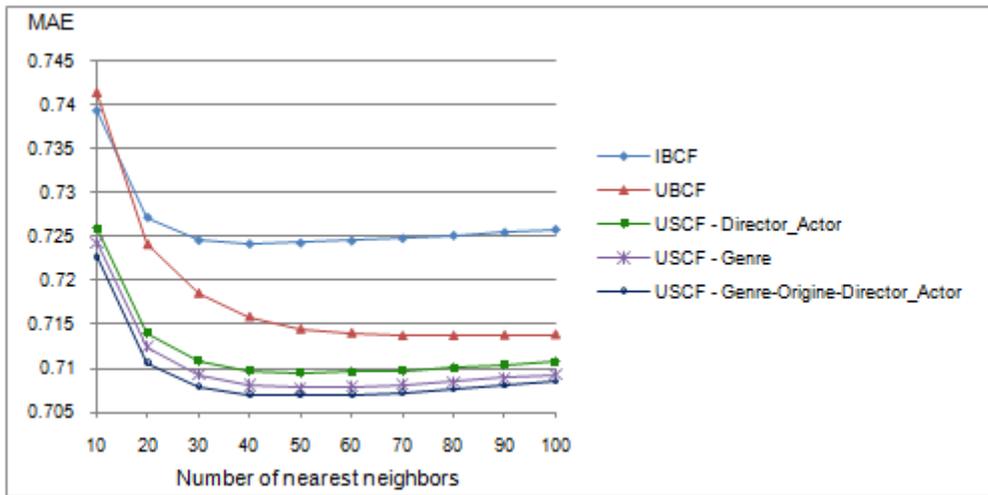


Figure 3: Evaluation of USCF against standards CF.

sis that all attributes don't have the same relevance to users. The attribute *origin* can be less significant in the choice of users than the *genre*, *actor* or *director*, which is intuitively understandable.

6.3 Comparative Results of USCF against Standard CF Systems

Figure 3 depicts the recommendation accuracy of User Semantic Collaborative Filtering (USCF) in contrast to standard Item-Based CF (IBCF) and User-Based CF (UBCF). USCF-*<Attributes>* in plot means the list of relevant attributes involved in building the user semantic model Q . For each relevant attribute, the suited algorithm is applied. So, Fuzzy CMean for *genre*, KMean for *origin*, and Rocchio with LSA (rank=1200) for combined attribute *direc-*

tor_actor. Furthermore, MAE has been plotted with respect to the number of neighbors (similar users) in the k-nearest-neighbor algorithm. In all cases, the MAE converges between 60 and 70 neighbors. Our approach, USCF (in plot) results in an overall improvement in accuracy for all attributes. In addition, the best performance is achieved by the combination *genre-origin-director_actor*. This improvement can be explained by many reasons. First, taking into account the semantic profile of items in a CF recommendation process. Second, for non dependent attribute, user semantic model is built according to a collaborative principle; ratings of all users are used to compute the semantic profile of each user. It is not the case of the *Average* algorithm; this may explain its results despite taking into account the semantic aspect. Third, the choice of the attribute can have significant influence on improving the accuracy. Lastly, users seman-

tic model Q has few missing values, so, it allows inferring similarity between two given users even when they have any items rated in common.

7 CONCLUSION AND FUTURE WORK

The approach presented in this paper is a component of a global work, which the aim, is to semantically enhanced collaborative Filtering recommendation and to resolve the scalability problem by reducing the dimension. For this purpose, we have designed a new hybridization technique, which predicts users' preferences for items based on their inferred preferences for semantic information. We have defined two classes of attributes: *dependent* and *non dependent* attribute, and presented a suited algorithm for each class for building user semantic attribute model. The aim of this paper is to present our approach for building user semantic attribute model for dependent attribute. We have defined an algorithm based on Rocchio algorithm and have applied Latent Semantic Analysis (LSA) for dimension reduction. Our approach provides solutions to the scalability problem, and alleviates the data sparsity problem by reducing the dimensionality of data. The experimental results show that USCF algorithm improves the prediction accuracy compared to usage only approach (UBCF and IBCF) and hybrid algorithm (Average). In addition, we have shown that applying Rocchio formula on non dependent attribute, decreases significantly the prediction accuracy compared to results obtained with machine learning algorithms. Furthermore, we have experimentally shown that all attributes don't have the same importance to users. Finally, experiments have shown that the combination of relevant attributes enhances the recommendations.

An interesting area of future work is to use machine learning techniques to infer relevant attributes. We will also study the extension of the user semantic model to non structured data in witch items are described by free text. Lastly, study how our approach can provide solution to the cold start problem in which new user has few ratings. Indeed, CF cannot provide recommendation because similarities with others users cannot be computed.

REFERENCES

- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6):734–749.
- Balabanovic, M. and Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Commun. ACM*, 40(3):66–72.
- Ben Ticha, S., Roussanaly, A., and Boyer, A. (2011). User semantic model for hybrid recommender systems. In *The 1st Int. Conf. on Social Eco-Informatics - SOTICS*, Barcelona, Espagne. IARIA.
- Ben Ticha, S., Roussanaly, A., Boyer, A., and Bsaïes, K. (2012). User semantic preferences for collaborative recommendations. In *13th Int. Conf. on E-Commerce and Web Technologies - EC-Web*, pages 203–211, Vienna, Austria. Springer.
- Burke, R. D. (2007). Hybrid web recommender systems. In *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 377–408. Springer.
- Dumais, S. T. (2004). Latent semantic analysis. *Annual Review of Information Science and Technology*, 38(1):188–230.
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53.
- HetRec2011 (2011). In *2nd Int Workshop on Information Heterogeneity and Fusion in Recommender Systems*. The 5th ACM Conf. RecSys.
- Lops, P., de Gemmis, M., and Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, pages 73–105. Springer US.
- Manzato, M. G. (2012). Discovering latent factors from movies genres for enhanced recommendation. In *The 6th ACM conf. on Recommender systems - RecSys*, pages 249–252, Dublin, Ireland.
- Mobasher, B., Jin, X., and Zhou, Y. (2003). Semantically enhanced collaborative filtering on the web. In *1st European Web Mining Forum*, volume 3209, pages 57–76, Cavtat-Dubrovnik, Croatia.
- Pazzani, M. and Billsus, D. (2007). Content-based recommendation systems. In *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 325–341. Springer Berlin Heidelberg.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In *The 1994 ACM conf. on Computer supported cooperative work*, pages 175–186, Chapel Hill, North Carolina, USA.
- Rocchio, J. (1971). Relevance feedback in information retrieval. In Salton, G., editor, *The Smart Retrieval System - Experiments in Automatic Document Processing*, chapter 14, pages 313–323. Prentice-Hall, Inc.
- Salton, G. (1989). *Automatic Text Processing*. Addison-Wesley.
- Sarwar, B. M., Karypis, G., Konstan, J. A., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *The 10 Int. WWW Conf.*, pages 285–295, Hong Kong, China.
- Sen, S., Vig, J., and Riedl, J. (2009). Tagommenders: connecting users to items through tags. In *The 18th Int Conf. on WWW*, pages 671–680, Madrid, Spain.
- Su, X. and Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Adv. Artificial Intelligence*.

SHORT PAPERS

Extracting Multi-item Sequential Patterns by Wap-tree Based Approach

Kezban Dilek Onal and Pinar Karagoz

Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
{dilek, karagoz}@ceng.metu.edu.tr

Keywords: WAP-Tree (Web Access Pattern Tree), Sequential Pattern Mining, FOF (First Occurrence Forest), Sibling Principle, Web Usage Mining.

Abstract: Sequential pattern mining constitutes a basis for solution of problems in web mining, especially in web usage mining. Research on sequence mining continues seeking faster algorithms. WAP-Tree based algorithms that emerged from the web usage mining literature have shown a remarkable performance on single-item sequence databases. In this study, we investigate the application of WAP-Tree based mining to multi-item sequential pattern mining and we present MULTI-WAP-Tree, which extends WAP-Tree for multi-item sequence databases. In addition, we propose a new algorithm MULTI-FOF-SP (MULTI-FOF-Sibling Principle) that extracts patterns on MULTI-WAP-Tree. MULTI-FOF-SP is based on the previous WAP-Tree based algorithm FOF (First Occurrence Forest) and an early pruning strategy called "Sibling Principle" from the literature. Experimental results reveal that MULTI-FOF-SP finds patterns faster than PrefixSpan on dense multi-item sequence databases with small alphabets.

1 INTRODUCTION

Sequential pattern mining is one of the major tasks in data mining and it constitutes a basis for solution of pattern discovery problems in various domains (Mooney and Roddick, 2013). Use of sequential pattern mining in web usage mining problems helps discovering user navigation patterns on web sites that can guide processes like recommendation and web site design.

In some applications of sequence mining like web usage mining and bioinformatics, the sequences have fixed transaction size of 1. This specific case of sequence mining is referred as *Single-Item Sequential Pattern Mining*. In accordance with this naming, the general form of the problem is referred as *Multi-Item Sequential Pattern Mining*.

WAP-Tree (Web Access Pattern Tree) based algorithms have shown remarkable execution time performance on single-item sequential pattern mining (Mabroukeh and Ezeife, 2010). WAP-Tree is a compact data structure for representing single-item sequence databases (Pei et al., 2000). There is a considerable number of WAP-Tree based algorithms in the literature. Among the previous WAP-Tree based algorithms, PLWAP (Ezeife and Lu, 2005) is reported to outperform well known general sequential pattern mining algorithms PrefixSpan and LAPIN

on single-item sequence databases (Mabroukeh and Ezeife, 2010) for single item patterns.

Inspired by the success of WAP-Tree, we designed a new data structure MULTI-WAP-Tree which extends WAP-Tree for representing multi-item/general sequence databases. Secondly, we propose a new sequential pattern mining algorithm MULTI-FOF-SP based on MULTI-WAP-Tree inspired by the WAP-Tree based algorithm FOF (First Occurrence Forest) (Peterson and Tang, 2008). MULTI-FOF-SP, integrates FOF approach and the early pruning idea "Sibling Principle" from the previous studies (Masseglia et al., 2000) and (Song et al., 2005).

We have analyzed the performance of the proposed method on several test cases. The results show that MULTI-WAP-Tree and the associated mining algorithm on this structure presents successful results, especially for dense multi-item sequence databases with small alphabets.

The rest of the paper is composed of four sections. In Section 2, we provide background information on WAP-Tree and the FOF algorithm. We present our contributions, MULTI-WAP-Tree and MULTI-FOF-SP, in Section 3 and Section 4, respectively. Finally, we present experiment results in Section 5 and conclude in Section 6.

2 RELATED WORK AND BACKGROUND

Sequential pattern mining is the extraction of the sequences that occur at least as frequently as the minimum support $minSupport$ in a sequence database D (Agrawal and Srikant, 1995). The sequential pattern mining algorithms in the literature can be reviewed under four categories (Mabroukeh and Ezeife, 2010), namely Apriori based, vertical projection, pattern growth and early pruning.

The WAP-Tree based algorithms follow the pattern growth approach on single-item sequential pattern mining. Since our study is focused on WAP-Tree based mining, we first review the WAP-Tree data structure in the next subsection. Secondly, we summarize the pattern growth approach in Subsection 2.2. Finally, we review the FOF (First Occurrence Forest) algorithm in comparison with the other WAP-Tree based algorithms in Subsection 2.3.

2.1 WAP-Tree

The WAP-Tree (Web Access Pattern Tree) is a tree data structure that is designed to represent a single-item sequence database. This data structure was first introduced together with the WAP-Mine algorithm (Pei et al., 2000). WAP-Mine algorithm converts the sequence database into a WAP-Tree with an initial database scan and performs mining on the WAP-Tree to find frequent patterns.

Table 1: Sample Single-item Sequence Database.

| Sequence Id | Sequence |
|-------------|----------|
| 1 | aba |
| 2 | adcdb |
| 3 | beae |
| 4 | ac |

Figure 1 illustrates the WAP-Tree for the sequence database given in Table 1 under minimum support 0.5. Each node of the WAP-Tree comprises two fields: Item and Count. Count field of a node n stores the count of the sequences starting with the prefix obtained by following the path from root to n . The two children of R , $(a:3)$ and $(b:1)$ in Figure 1 indicate that there are 3 sequences starting with a and a single sequence starting with b . WAP-Tree provides a compact representation since shared prefixes can be encoded on the nodes.

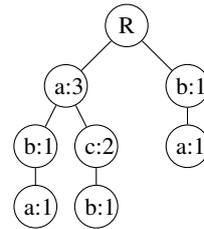


Figure 1: WAP-Tree For The Sequence Database in Table 1 Under Support Threshold 0.5.

2.2 Pattern Growth Approach

The pattern growth approach follows the divide and conquer paradigm (Han et al., 2005). The original problem of finding the frequent sequences is recursively broken down into smaller problems by shrinking the database into projected databases. The lexicographic search space is traversed depth-first and whenever a pattern is found to be frequent, the database is projected by the pattern. The mining process is recursed on the projected database.

The realization of the pattern growth approach depends on the pattern growing direction, how the projected databases are represented and located during mining (Han et al., 2005). There are two alternative directions for growing patterns: namely suffix growing and prefix growing. The patterns are grown by appending symbols in prefix growing whereas they are grown by prepending symbols in suffix growing.

| $D _{\epsilon}$ | $D _a$ | $D _{ab}$ |
|------------------|---------|------------|
| aba | ba | a |
| adcdb | dcdb | ϵ |
| beae | e | ϵ |
| ac | c | ϵ |

Figure 2: Projected Databases for Different Patterns.

Three sample projected databases for prefix-growing are given in horizontal database representation in Figure 2. From left to right, the tables in Figure 2 are the projected databases of the patterns ϵ, b, ba in the sequence database D given in Table 1. The projections of the sequences can be traced by following the rows of the tables at the same level. For example, the a -projection of the sequence aba is ba . The b -projection of the sequence ba is a and it is equal to the ab -projection of aba .

2.3 FOF (First Occurrence Forest)

FOF is a prefix growing WAP-Tree based algorithm. The FOF algorithm represents the sequence database as a WAP-Tree and it adopts the recursive projected

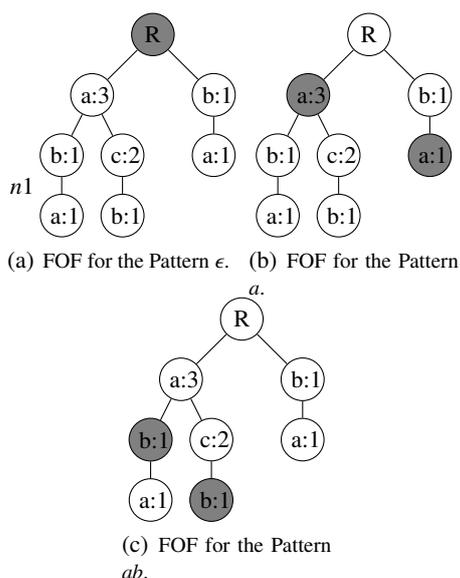


Figure 3: FOFs for Prefix Growing On WAP-Tree.

database mining approach of the pattern growth algorithms. The FOF algorithm represents the projected databases as First Occurrence Forests (FOFs). First Occurrence Forest is a list of WAP-Tree nodes which root a forest of WAP-Trees. The WAP-Trees rooted by the FOF nodes of a pattern encode the projections of the sequences by the pattern. FOF representation enables easy support counting for a pattern by summing the count values of the FOF nodes.

The mining process of the FOF algorithm is based on searching the first level occurrences of symbols recursively. At each pattern growing step by a symbol, the first level occurrences of the symbol in the forest of WAP-Trees defined by the FOF of the original pattern are assigned as the FOF nodes for the grown pattern. The mining process is recursed on the FOF nodes for the grown pattern.

The mining process of the FOF algorithm on the WAP-Tree in Figure 1 is partially illustrated in Figure 3. FOF of three different patterns are presented in Figure 3. The shaded nodes in the WAP-Trees indicate the FOF nodes. Initially, the FOF for the pattern ϵ is $\{R:4\}$. Secondly, the first level occurrences of the symbol a under the node $R:4$ are found as the FOF for the pattern a . The node n_1 is not included in the FOF of the symbol a since it is not a first level occurrence. In the next level of recursion, the FOF algorithm mines the FOF for a . The FOF for ab is the set of first level b occurrences in the sub-trees rooted by the shaded nodes for pattern a . FOF of ab is found by searching the first level b occurrences in the sub-trees rooted by the FOF nodes for the pattern a .

The projected database representation as a list of nodes, i.e. FOF, was adapted by previous WAP-

Tree based algorithms except WAP-Mine. All WAP-Tree based algorithms in the literature follow pattern-growth approach. However, they differ in the pattern growing direction. WAP-Mine does suffix growing and represents projected databases as WAP-Trees. All of the latter algorithms, namely PLWAP (Ezeife and Lu, 2005), FLWAP (Tang et al., 2006), FOF (Peterson and Tang, 2008) and BLWAP (Liu and Liu, 2010), do prefix growing.

Prefix growing WAP-Tree based algorithms differ in their approach for locating the first level occurrences of a symbol in a WAP-Tree. The FOF algorithm locates first level occurrences of an item by simple depth-first search. On the contrary, PLWAP, BLWAP and FLWAP leverage additional data structures *links* and *header table* besides the WAP-Tree. The links connect all occurrences of an item in a WAP-Tree and the links can be followed starting from the header table. The first level occurrences of an item in a WAP-Tree can be easily found by following the links of an item and filtering the occurrences which are found at the first level. The FOF algorithm is reported to outperform PLWAP and FLWAP in (Peterson and Tang, 2008) although. FOF uses less memory since it mines WAP-Tree without additional structures. Besides, although the links provide direct access to the occurrences, filtering the first level occurrences brings an extra cost to PLWAP and FLWAP.

3 MULTI-WAP-TREE

MULTI-WAP-Tree is an extended WAP-Tree which can represent both single and multi-item sequence databases. MULTI-WAP-Tree is identical to the WAP-Tree in that it contains only frequent items in its nodes and each sequence in the sequence database is encoded in MULTI-WAP-Tree on a path from the root to a node of the tree.

MULTI-WAP-Tree differs from the WAP-Tree in two points:

- MULTI-WAP-Tree has two types of edges between nodes : *S-Edge* and *I-Edge* whereas WAP-Tree has a single edge type.
- MULTI-WAP-Tree nodes keep a pointer to their parent nodes in addition to the fields of WAP-Tree.

MULTI-WAP-Tree is able to represent a multi-item sequence database owing to two different edge types between nodes. A multi-item sequence is composed of a series of item-sets. WAP-Tree is not able to represent multi-item databases since it cannot express boundaries between item-sets in multi-item se-

quences. S-Edges of MULTI-WAP-Tree can encode item set boundaries. An S-Edge from a node to its child indicates the separator between two item sets: item set ending with the parent and item set starting with the child. On the contrary, the nodes connected with I-Edges are always in the same item set.

Figure 4(d) shows the MULTI-WAP-Tree for the sample multi-item database in Table 2 under the support threshold 0.5. The dashed edges in the figure are I-Edges, whereas the plain edges are S-Edges. The prefix represented by the gray coloured node is (ab) . Although both of the edges originating from this node point to c labeled nodes, they yield different prefixes since their edge types are different. The child of this node connected with an S-Edge represents the prefix $(ab)(c)$, whereas the other child connected with the I-Edge represents the prefix (abc) .

Table 2: Sample Multi-item Sequence Database.

| Sequence Id | Sequence |
|-------------|-------------|
| 1 | $(ab)(c)$ |
| 2 | $(a)(b)(c)$ |
| 3 | $(abc)(c)$ |

The second extension to the WAP-Tree, "pointer to parent node" in MULTI-WAP-Tree nodes, enables tracking item sets upwards in the tree in the mining phase. This additional field does not contribute to the database representation but it is an important construct for mining MULTI-WAP-Tree.

4 MULTI-FOF-SP

MULTI-FOF-SP algorithm is a multi-item sequential pattern mining algorithm based on the representation MULTI-WAP-Tree. MULTI-FOF-SP algorithm has three basic steps, as in all WAP-Tree based algorithms:

1. Scan database to find frequent items.
2. Scan database and build MULTI-WAP-Tree.
3. Mine frequent patterns from MULTI-WAP-Tree.

4.1 Building MULTI-WAP-Tree

For MULTI-WAP-Tree construction, after eliminating the infrequent items, each sequence is inserted into the tree starting from the root, updating counts of shared prefix nodes and inserting new nodes for the unshared suffix part, as in WAP-Tree construction. While constructing a WAP-Tree, checking only equality of item fields of nodes is sufficient. However, in the MULTI-WAP-Tree case, checking the equality

of edge types is also required. To illustrate, Figure 4 depicts the MULTI-WAP-Tree database construction algorithm on the mini database in Table 2. When inserting the second sequence $(a)(b)(c)$, although the a node already has a b child, since this child is connected with an I-Edge, a new b node is added with an S-Edge.

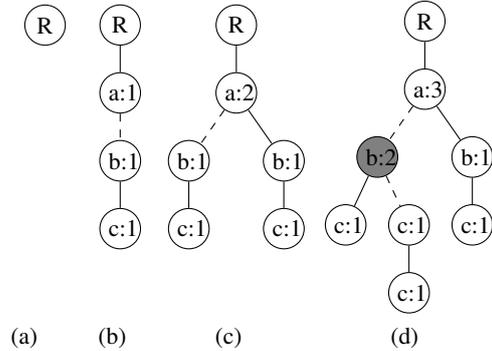


Figure 4: Building Steps of the MULTI-WAP-Tree for the Database in Table 2. Left to right: MULTI-WAP-Tree after sequences $(ab)(c)$, $(a)(b)(c)$, $(abc)(c)$ are inserted successively.

4.2 Mining MULTI-WAP-Tree

MULTI-FOF-SP follows prefix growing and FOF approach for extracting multi-item sequential patterns on MULTI-WAP-Tree. MULTI-FOF-SP represents projected databases in FOF form and finds first level occurrences of an item by simple depth first search on the MULTI-WAP-Tree, as in FOF algorithm. However, MULTI-FOF-SP algorithm differs from FOF in two points which are explained in detail in the following two subsections.

4.2.1 S-Occurrence vs. I-Occurrence

Existence of two different edge types S-Edge and I-Edge in MULTI-WAP-Tree necessitates distinction between S-Occurrences and I-Occurrences. S-Occurrences are the first level occurrences that yield a sequence extension, whereas I-Occurrences yield an item-set extension.

MULTI-FOF-SP algorithm treats S-Occurrences and I-Occurrences as occurrences of different symbols and searches for them separately. Whenever an occurrence is located, it is subjected to a test for its occurrence type. If an occurrence is found in the subtree rooted by $startNode$, the algorithm decides on the type of occurrence based on the three rules below:

1. A node is an S-Occurrence if there exists at least one S-Edge on the path from the $startNode$ to this node.

2. A node is an I-Occurrence if there are only I-Edges on the path from the *startNode* to this node.
3. A node is an I-Occurrence if the last itemset of the grown pattern can be found by following the ancestor nodes of the node before an S-Edge is encountered.

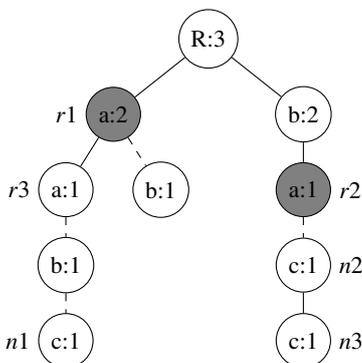


Figure 5: Find First Occurrences Illustration.

It is crucial to note that a node can be both an S-Occurrence and I-Occurrence at the same time. To illustrate the rules above, consider the MULTI-WAP-Tree in Figure 5. The gray shaded nodes $r1$ and $r2$ represent the FOF for the pattern (a) . There exists three c nodes, namely $n1$, $n2$ and $n3$, under this FOF. There are two S-Occurrences $\{n1, n3\}$ of c and two I-Occurrences $\{n1, n2\}$ of c under the FOF for the pattern (a) . I-Occurrences contribute to support count of (ac) , whereas S-Occurrences contribute to that of $(a)(c)$.

$n2$ is an I-Occurrence according to Rule 1 since there exists no S-Edges between $n2$ and its ancestor $r2$. On the contrary, $n3$ is an S-Occurrence since there exists an S-Edge on the route from $n3$ to $r2$. Finally, $n1$ is both an S-Occurrence and I-Occurrence since it matches both Rule 1 and Rule 3. $n1$ is an S-Occurrence because there exists an S-Edge between $n1$ and $r1$. In addition, $n1$ is an I-Occurrence since the last item-set of the grown pattern, namely (ac) , can be obtained by backtracking with only I-Edges from $n1$ to the ancestor node $r3$. The backtracking operation is performed using the pointers to the parent nodes which were mentioned as a difference from the WAP-Tree previously.

4.2.2 Sibling Principle

Sibling principle is an early pruning idea which was used in the previous studies (Song et al., 2005) and (Masseglia et al., 2000). It is an expression of the Apriori principle (Agrawal and Srikant, 1995) on the lexicographic search tree. According to the Apriori

principle, the sequences can be judged as infrequent if any of its sub-sequences is known to be infrequent. This principle can be modeled in lexicographic tree considering the siblings of a frequent pattern. Sibling principle requires checking sibling nodes of a node in the lexicographic tree and imposes constraints on the set of grown patterns. If a sibling node s of a node n is not frequent, a sequence which is a super-sequence of both s and n is pruned.

Figure 6 shows the subspace of the lexicographic search space processed by the MULTI-FOF-SP algorithm during mining the database in Table 2. The dashed edges in the tree indicate item-set extensions whereas the normal edges correspond to sequence extensions. The \checkmark sign indicates the sequence is frequent. All the patterns represented by the nodes except the underlined nodes are subjected to support counting by the algorithm MULTI-FOF-SP during mining the WAP-Tree given in Figure 4(d). The underlined nodes encode the sequences that are pruned owing to the sibling principle by MULTI-FOF-SP algorithm. For instance, the leftmost c node in the tree which represents the pattern (abc) can be early pruned by the sibling principle since the sibling pattern (ac) is found to be infrequent.

The numbers on the nodes indicate the order of traverse by MULTI-FOF-SP during mining. It is crucial to note that, the search space is traversed with a hybrid traversal strategy in order to apply the sibling principle. MULTI-FOF-SP combines the depth-first traversal of pattern growth approach and the breadth first approach imposed by the Apriori principle.

5 EXPERIMENTS

In this section, we present the experiments we have conducted in order to evaluate performance of the algorithm MULTI-FOF-SP. We compared execution time and memory usage performance of MULTI-FOF-SP on multi-item sequence databases with the algorithms PrefixSpan¹ and LAPIN-LCI².

We generated several synthetic sequence databases using IBM Quest Data Generator (Agrawal et al., 1993). We downloaded the IBM Quest Data Generator executable in Illimine Software Package Version 1.1.0 from the web site³. In order to obtain

¹We downloaded PrefixSpan executable in Illimine Software Package Version 1.1.0 from web site of Illimine Project <http://illimine.cs.uiuc.edu/download/>.

²We downloaded LAPIN-LCI executable from <http://www.tkl.iis.u-tokyo.ac.jp/yangzl/soft/LAPIN/index.htm>

³<http://illimine.cs.uiuc.edu/download/>

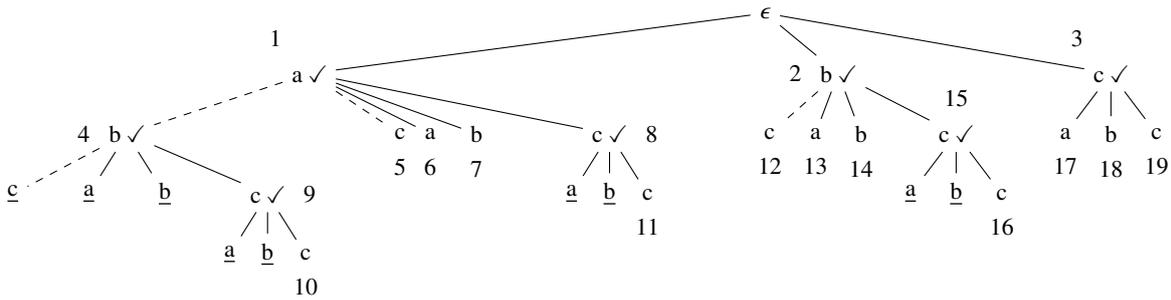


Figure 6: The Subspace of the Lexicographic Search Space Processed by the MULTI-FOF-SP Algorithm for Mining the Database in Table. 2

a comprehensive data set, we determined a set of values for each of the parameters C (Average length of sequences) = {25}, T (average size of itemsets) = {3,7}, D (number of sequences) = {200K,800K} and N (size of database alphabet) = {10,500}. We generated databases with combination of values from these sets.

Each sequence database in the experiment set is named in accordance with the parameter values. For instance C25T3S25I3N10D200k specifies a database generated with parameters $C=25$, $T=3$, $S=25$, $I=3$, $N=10$ and $D=200K$. The support values for the experiment sets are chosen such that comparative performance results can be obtained in a reasonable time.

We performed experiments on a personal computer with Intel(R) Core(TM) i7 860 @2.80Ghz 2.80 Ghz CPU, 8 GB installed memory and Windows 7 Professional 64-bit operating system. For each experiment case identified by the 4-tuple (Algorithm, Sequence Database, MinSupport), we report the execution time and memory consumption of algorithm on the sequence database. We repeated each test case run at least three times and reported the average value for all the algorithms. Due to space limitation, we present results of only a part of the conducted experiments.

As the first set of experiments, we present results on two different sequence databases of alphabet size $N=500$ in Figure 7 and Figure 8. In these experiments with large alphabet, PrefixSpan is faster than MULTI-FOF-SP in all of the cases. Although, LAPIN-LCI is faster than MULTI-FOF-SP in some cases, there are also cases in which LAPIN ends unexpectedly consuming more than 2 GBs of memory. MULTI-FOF-SP can prune the search space owing to sibling principle yet it may not be sufficient when the alphabet is large.

As the second set of experiments, we present results on three different sequence databases of alphabet size $N=10$ in Figure 9 and Figure 10.

MULTI-FOF-SP outperforms PrefixSpan in terms of execution time on all sequence databases with alphabet size $N=10$. In most of the cases, MULTI-

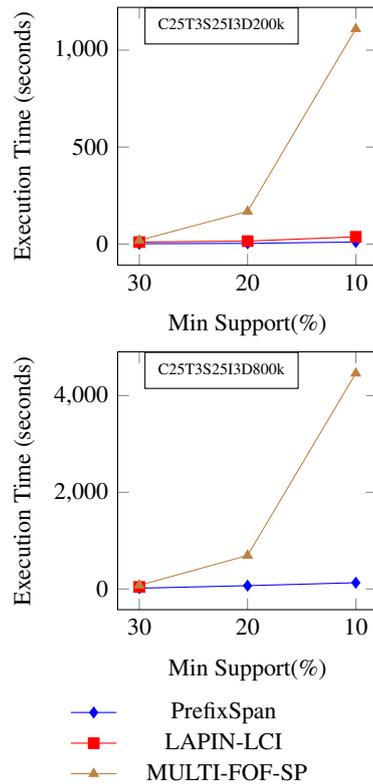


Figure 7: Execution Times of Algorithms on Databases with $N=500$.

FOF-SP is faster than LAPIN-LCI yet there are cases in which LAPIN-LCI is slightly faster. However, there are also cases LAPIN-LCI could not complete execution since it ran out of memory. One such case is given in Figure 10 for the database C25T7S25I7N10D800k when the minimum support value is decreased to 99.3.

MULTI-FOF-SP ranks the second on databases with $N=10$ in terms of memory consumption. The algorithm stores many FOFs in the memory in order to apply sibling principle. However, the memory consumption is moderate because this cost is compen-

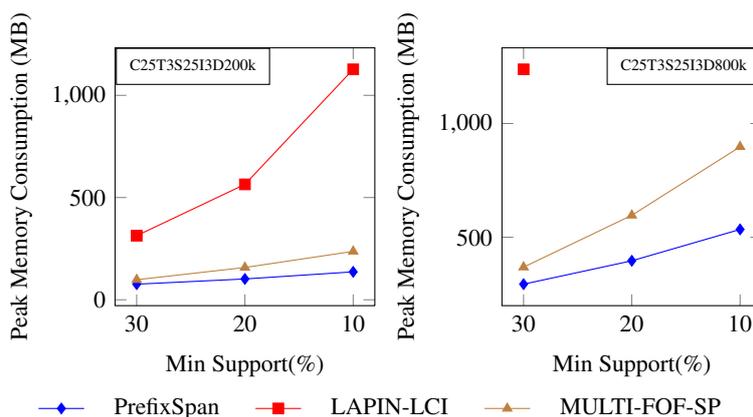


Figure 8: Memory Consumption of Algorithms on Databases with N=500.

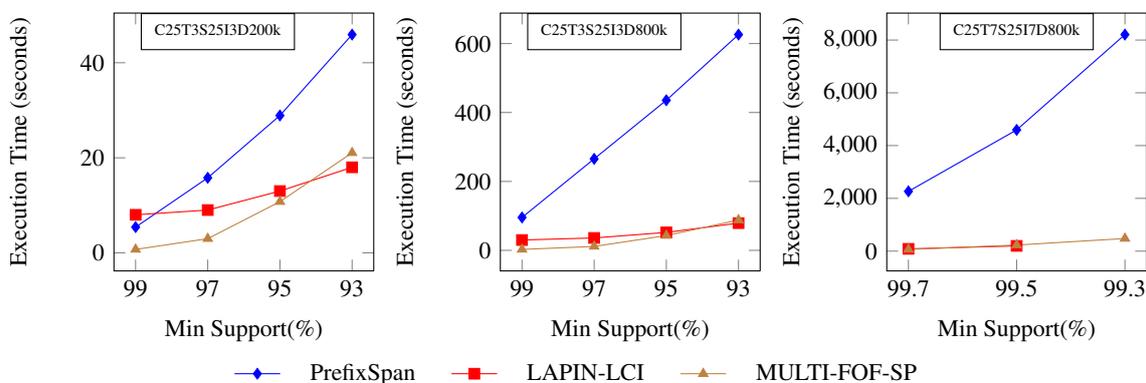


Figure 9: Execution Times of Algorithms on Databases with N=10.

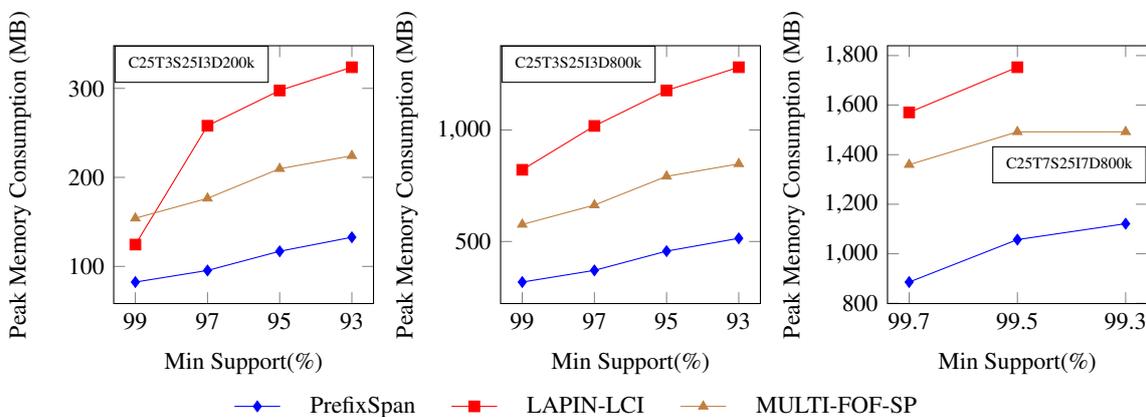


Figure 10: Memory Consumption Of Algorithms on Databases with N=10.

sated by the compression provided by the MULTI-WAP-Tree data structure.

MULTI-FOF-SP is faster on the databases with smaller alphabets owing to the compression provided by the MULTI-WAP-Tree. For instance, MULTI-WAP-Tree has at most 20 child nodes of the root regardless of the number of sequences in the database

when the alphabet size is 10. Moreover, the width of the WAP-Tree is never larger than the number of sequences. However, it is crucial to note that it is more efficient to mine the projected database in horizontal representation than mining the tree structure. Scanning the tree requires tracking the edges between the nodes. Besides, WAP-Tree node occupies more

space than the space occupied by an item in horizontal representation. Consequently, if the FOF approach were applied on the MULTI-WAP-Tree without the sibling principle, both memory and time requirements of FOF approach would be higher in cases of large alphabets. A high degree of compression by the MULTI-WAP-Tree is required to outperform PrefixSpan and LAPIN in terms of both execution time and memory. We observed that this requirement cannot be met in case of large alphabets.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a new data structure MULTI-WAP-Tree and a new algorithm MULTI-FOF-SP for extracting multi-item sequence patterns. MULTI-WAP-Tree is the first tree structure for representing general sequence databases. MULTI-FOF-SP employs the early pruning idea Sibling Principle.

We have experimented on several test cases to compare MULTI-FOF-SP with previous multi-item sequence mining algorithms, PrefixSpan and LAPIN-LCI. Experiments revealed that MULTI-FOF-SP outperforms PrefixSpan and has a performance close to LAPIN-LCI in terms of execution time on dense multi-item databases with small alphabets. In addition, it has a better performance than LAPIN-LCI in terms of memory usage for these databases.

In this work, we devised a MULTI-WAP-Tree based algorithm that uses sibling principle and obtained good results. As a continuation of this line, other existing tree based algorithms can be investigated for multi-item sequence mining using the MULTI-WAP-Tree data structure.

REFERENCES

- Agrawal, R., Imelinski, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, pages 207–216. ACM.
- Agrawal, R. and Srikant, R. (1995). Mining sequential patterns. In *Proceedings of the Eleventh International Conference on Data Engineering (ICDE'95)*, pages 3–14. IEEE.
- Ezeife, C. and Lu, Y. (2005). Mining web log sequential patterns with position coded pre-order linked wap-tree. *Data Mining and Knowledge Discovery*, 10(1):5–38.
- Han, J., Pei, J., and Yan, X. (2005). Sequential pattern mining by pattern-growth: Principles and extensions*. In Chu, W. and Lin, T., editors, *Foundations and Advances in Data Mining*, volume 180 of *Studies in Fuzziness and Soft Computing*, pages 183–220. Springer Berlin Heidelberg.
- Liu, L. and Liu, J. (2010). Mining web log sequential patterns with layer coded breadth-first linked wap-tree. In *International Conference of Information Science and Management Engineering (ISME'2010)*, volume 1, pages 28–31. IEEE.
- Mabroukeh, N. and Ezeife, C. (2010). A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys (CSUR)*, 43(1):3.
- Masseglia, F., Poncelet, P., and Cicchetti, R. (2000). An efficient algorithm for web usage mining. *Networking and Information Systems Journal*, 2(5/6):571–604.
- Mooney, C. H. and Roddick, J. F. (2013). Sequential pattern mining – approaches and algorithms. *ACM Comput. Surv.*, 45(2):19:1–19:39.
- Pei, J., Han, J., Mortazavi-Asl, B., and Zhu, H. (2000). Mining access patterns efficiently from web logs. *Knowledge Discovery and Data Mining. Current Issues and New Applications*, pages 396–407.
- Peterson, E. and Tang, P. (2008). Mining frequent sequential patterns with first-occurrence forests. In *Proceedings of the 46th Annual Southeast Regional Conference (ACMSE)*, pages 34–39. ACM.
- Song, S., Hu, H., and Jin, S. (2005). Hvsm: A new sequential pattern mining algorithm using bitmap representation. In Li, X., Wang, S., and Dong, Z., editors, *Advanced Data Mining and Applications*, volume 3584 of *Lecture Notes in Computer Science*, pages 455–463. Springer Berlin Heidelberg.
- Tang, P., Turkia, M., and Gallivan, K. (2006). Mining web access patterns with first-occurrence linked wap-trees. In *Proceedings of the 16th International Conference on Software Engineering and Data Engineering (SEDE'07)*, pages 247–252. Citeseer.

Improving Opinion-based Entity Ranking

Christos Makris and Panagiotis Panagopoulos

Department of Computer Engineering and Informatics, University of Patras, Patras, Greece
{makri, panagoppa}@ceid.upatras.gr

Keywords: Opinion Mining and Sentiment Analysis, Web Information Filtering and Retrieval, Searching and Browsing.

Abstract: We examine the problem of entity ranking using opinions expressed in users' reviews. There is a massive development of opinions and reviews on the web, which includes reviews of products and services, and opinions about events and persons. For products especially, there are thousands of users' reviews, that consumers usually consult before proceeding in a purchase. In this study we are following the idea of turning the entity ranking problem into a matching preferences problem. This allows us to approach its solution using any standard information retrieval model. Building on this framework, we examine techniques which use sentiment and clustering information, and we suggest the naive consumer model. We describe the results of two sets of experiments and we show that the proposed techniques deliver interesting results.

1 INTRODUCTION

The rapid development of web technologies and social networks, has created a huge volume of reviews on products and services, and opinions on events and individuals.

Opinions are an important part of human activity because they affect our behavior in decision-making. It has become a habit for consumers to be informed by the reviews of other users, before they make a purchase of a product or a service. Businesses also want to be able to know the opinions concerning all their products or services and modify appropriately their promotion and their further development.

The consumer, however, in order to create an overall evaluation assessment for a set of objects of a specific entity, must refer to many reviews. From those reviews he must extract as many opinions as possible, in order to create an observable conclusion for each of the objects, and then to finally classify the objects and discern those that are notable. It is clear that this multitude of opinions creates a challenge for the consumer and also for the entity ranking systems.

Thus we recognize that the development of computational techniques, that help users to digest and utilize all opinions, is a very important and interesting research challenge.

In (Ganesan and ChengXiang, 2012) it is

depicted the setup for an opinion-based entity ranking system. The idea is that each entity is represented by the text of all its reviews, and that the users of such a system, determine their preferences on several attributes during the evaluation process. Thus we can expect that a user's query, will consist of preferences on multiple attributes. By turning the problem of assessing entities into a matching preferences problem, we can use, in order to solve it, any standard information retrieval model. Given a query from the user, which consists of keywords and expresses the desired characteristics an entity must have, we can evaluate all candidate entities based on how well the opinions of those entities match the user's preferences.

Building on this idea. in the present paper, we develop schemes which take into account clustering and sentiment information about the opinions expressed in reviews. We also propose a naive consumer model as a setup that uses information from the web to gather knowledge from the community in order to evaluate the entities that are more important.

2 RELATED WORK

In this study we deal with the problem of creating a ranked list of entities using users' reviews. In order

to approach effectively its handling, we are moving to the direction of aspect-oriented opinion mining or feature-based opinion mining as defined in (Ganesan and ChengXiang, 2012). In this consideration, each entity is represented as the total text of all the available reviews for it, and users express their queries as preferences in multiple aspects. Entities are evaluated depending on how well the opinions, expressed in the reviews, are matched user's preferences.

Regarding reviews, a great deal of research has been done on the classification of reviews to positive and negative based on the overall sentiment information contained (document level sentiment classification). There have been proposed several supervised in (Gamon, 2005), (Pang and Lee, 2004), unsupervised in (Turney and Littman, 2002), (Nasukawa and Yi, 2003), and also hybrid in (Pang and Lee, 2005), (Prabowo and Thelwall, 2009) techniques.

A related research area is opinion retrieval in (Liu, 2012). The goal of opinion retrieval is to identify documents that contain opinions. An opinion retrieval system is usually created on top of the classical recovery models, where relevant documents are initially retrieved and then some opinion analysis techniques are being used to export only documents containing opinions. In our approach we are assuming that we already have available the texts, which contain the opinions for the entities.

Another related research area is the field of Expert Finding. In this area, the goal is to recover one ranked list of persons, which are experts on a certain topic (Fang and Zhai, 2007), (Baeza-Yates and Ribeiro-Neto, 2011), (Wang et al., 2010). In particular, we are trying to export a ranked list of entities, but instead of evaluating the entities based on how well they match a topic, we use the opinions for the entities and we are observing how well they match the user's preferences.

Also, there has been much research in the direction of using reviews for provisioning aspect based ratings in (Wang et al., 2010), (Snyder and Barzilay, 2007). This direction is relevant to ours, because by performing aspect based analysis, we can extract the ratings of the different aspects from the reviews. Thus we can assess entities based on the ratings of the aspects, which are in the user's interests.

In section 6, we examine the naive consumer model as an unsupervised schema that utilizes information from the web in order to yield a weight of importance to each of the features used for

evaluating the entities. We choose to use a formula that has some resemblance to those used in item response theory (ITL), (Hambleton et al., 1991) and the Rasch model (Rasch), (Rasch, 1960/1980). Item response theory is a paradigm for the design, analysis, and scoring of tests, questionnaires, and similar instruments measuring abilities, attitudes, or other variables. The mathematical theory underlying Rasch models is a special case of item response theory. There are approaches in text mining that use the Rasch model and item response theory such as (Tikves et al., 2012), (He, 2013). However our approach differs in the chosen metrics and in the applied methodology and we do not use explicitly any of the modeling capabilities of these theories. For other different approaches that take aspect weight into account see (Liu, 2012) and more specifically (Yu et al., 2011) however our technique is simpler and fits into the framework presented in (Ganesan and ChengXiang, 2012).

2.1 Novelty in Contribution

In (Ganesan and ChengXiang, 2012), they presented a setup for entity ranking, where entities are evaluated depending on how well the opinions expressed in the reviews are matched against user's preferences. They studied the use of various state-of-the-art retrieval models for this task, such as the BM25 retrieval function (Baeza-Yates and Ribeiro-Neto, 2011), (Robertson and Zaragoza, 2009), and they also proposed some new extensions over these models, including query aspect modeling (QAM) and opinion expansion. With these extensions they were given the opportunity to classical information retrieval models to detect subjective information, i.e. opinions, that exist in review texts. More specifically, the opinion expansion introduced intensifiers and common praise words with positive meaning, placing at the top entities with many positive opinions. This expansion favoured texts, and correspondingly entities, with positive opinions on aspects, which is the goal. *However this approach does not impose penalties for negative opinions.*

We further improve this setup by developing schemes, which take into account sentiment (section 4) and clustering information (section 5) about the opinions expressed in reviews. We also propose the naive consumer model in section 6.

3 THE PROBLEM OF RANKING ENTITIES AS INFORMATION RETRIEVAL PROBLEM

Consider an entity ranking system, RS, and a collection of entities $E = \{e_1, e_2, \dots, e_n\}$ of the same kind. Assume that each of the entities in the set E , is accompanied by a big collective text with all the reviews for it, written by some reviewers. Let $R = \{r_1, r_2, \dots, r_n\}$ be the set of all those texts. Then there exists an "1-1" relationship between the entities of E and the texts of R . Given a query q that is composed by a subset of the aspects, the RS system produces a ranking list of entities in E .

The idea to assess the entities, is to represent each entity with the text of all the reviews referred to in that entity. Given a keyword query by a user, which expresses the desirable features that an entity should have, we can evaluate the entities based on how well the review texts (r_i) match the user's preferences. So the problem of entity ranking becomes an information retrieval problem. Thus we can employ some of the known information retrieval models, such as BM25. This setup is being presented in (Ganesan and ChengXiang, 2012). In particular they employed the BM25 retrieval function (Baeza-Yates and Ribeiro-Neto, 2011), (Robertson and Zaragoza, 2009), the Dirichlet prior retrieval function (Zhai and Lafferty, 2001), and the PL2 function (Amati, and van Rijsbergen, 2002) and proposed some new extensions over these models, including query aspect modeling (QAM) and opinion expansion and they performed a set of experiments depicting the superiority of their approach. The QAM extension uses each query aspect to rank entities and then aggregates the ranked results from the multiple aspects of the query using an aggregation function such as the average score. The opinion expansion extension, expands a query with related opinion words found in an online thesaurus. The results of the experiments showed that while all three state-of-the-art retrieval models show improvement with the proposed extensions, the BM25 retrieval model is most consistent and works especially well with these extensions.

4 OPINION-BASED ASPECT RATINGS

Addressing the entity ranking problem as a matching preferences problem on specific features using an information retrieval model as presented in

(Ganesan and ChengXiang, 2012) favors texts, and correspondingly entities, with positive opinions on aspects, which is the goal. *However this approach does not impose penalties for negative opinions.* Then we seek to examine the performance of known techniques for sentiment analysis. These techniques take into account the positive and negative opinions on the entity rating process. Our goal is to compare their performance with the performance of the information retrieval approach.

Instead of using the significance of the features (aspects) of the query for a review text (r_i) to create a ranking of the review texts, we attempt to use the sentiment information that exists in the opinions, expressed in the review texts, on specific features. In order to create a model which takes into account the sentiment information of the opinions that are expressed in the reviews by the reviewers, we use two simple unsupervised sentiment analysis techniques.

Given that each review text r_i contains the users' opinions for a particular entity, we apply simple aspect-based sentiment analysis techniques to extract the sentiment information about the features from the sentences, and aspect-based summarization (or feature-based summarization) to calculate the score of the features throughout the text.

4.1 Lexicon-based Sentiment Analysis

Let $A = \{a_1, a_2, \dots, a_m\}$ be the set of the query aspects. We perform aspect level sentiment analysis, by extracting from the reviews r_i the polarity, $s(a_j)$, which is expressed for each of the aspect query keywords.

The total score the review r_i receives is the sum of the aspects sentiment scores, in this text, normalized by the number of aspects in the query. To calculate the sentiment score $s(a_j)$, we locate in the review text r_i the sentences on which any of the query aspects (a_j) appear and we assign to them a sentiment rating. The score $s(a_j)$ is the sum of the sentences scores on which there is the aspect a_j . To calculate the sentiment score of a sentence, we apply a pos tagging process (Tsuruoka), in order to tag every term with a pos tag, and we process only the terms that have been assigned the following pos tags (see List of part-of-speech tags at references):

{ RB / RBR / RBS / VBG / JJ / JJR / JJS }

as elements that usually contain sentiment information. For each of those terms we find the word's sentiment score in a sentiment dictionary (Liu, Sentiment Lexicon), 1 if it is labeled as a

positive concept term, -1 if it is labeled as a negative concept term. Finally we sum the scores of all the terms. If the sum is positive the sentence's sentiment score is 1, while if the sum is negative the sentence's sentiment score is -1.

Also we take care of the negation and we use sentiments shifters. If in a sentence there is one of the following words: {not, don't, none, nobody, nowhere, neither, cannot}, we reverse the polarity of the sentence's final sentiment score.

4.1.1 Query Expansion

This automatic process reads every review text (r_i), sentence by sentence, and processes only those that contain one or more of the query's aspect keywords. But users usually use different words or phrases to describe their opinions on a feature (aspect) of the entity. To manage this effect we perform query expansion on the original query, which we seek to enrich with synonyms of the aspects $\sigma(a_i)$, as they are from the semantic network WordNet (Fellbaum, 1998).

For example, suppose a query q which consists of the aspect keywords $\{a_1, a_2, a_3\}$. For each keyword in q , we try to find synonymous terms $\sigma(a_j)$ using the semantic network WordNet and we import them to the query. The final query that emerges is $q = (a_1, \sigma_{1a1}, \sigma_{2a1}, a_2, \sigma_{1a2}, \sigma_{2a2}, \sigma_{3a2}, a_3, \sigma_{1a3})$. In this case the sentiment score of the aspect a_i , $s_{exp}(a_i)$, is the sentiment score of the term a_i plus the sentiments scores of all imported terms σ_{jai} , $s(\sigma_{jai})$.

$$s_{exp}(a_i) = s(a_i) + \sum_{j=1,2,\dots,h} s(\sigma_{jai})$$

4.2 Syntactic Patterns based Sentiment Analysis

In this scheme we employ as base of our construction the algorithm that is presented in (Turney, 2002) in order to calculate the sentiment score of each sentence. This process performs analysis in a similar manner to the first. Like the first sentiment analysis technique, which is presented in section 4.1 above, so this technique reads every review text (r_i), sentence by sentence, and processes only those that contain one or more of the query's aspect keywords. However here, instead of using the sentiment score of the words in the sentence, we use the sentiment orientation (SO) of syntactic patterns in the sentence, which are usually used to form an opinion.

Syntactic patterns are identified within a sentence based on pos tags of terms, which appear in

a specific order. The following are syntactic patterns that are used to extract two-word phrases:

| 1st Word | 2nd Word | 3rd Word(not extracted) |
|------------|----------------|-------------------------|
| JJ | NN/NNS | anything |
| RB/RBR/RBS | JJ | not(NN/NNS) |
| NN/NNS | JJ | not(NN/NNS) |
| RB/RBR/RBS | VB/VBD/VBN/VBG | anything |

In order to calculate the sentiment orientation (SO) of the phrases, we use the point wise mutual information (PMI). The PMI metric measures the statistical dependence between two terms. The sentiment orientation of a phrase is calculated based on its relationship with a set of positive reference words and a set of negative reference words. We use the set '+' = {excellent, good} as positive reference words and the set '-' = {horrible, bad} as negative reference words, and we enrich these sets with synonyms from the semantic network WordNet. Thus the sentiment, orientation of a phrase is calculated as follows:

$$SO(\text{phrase}) = \log_2 \frac{\text{hits}(\text{phrase NEAR}^+)' \text{hits}(' - ')}{\text{hits}(\text{phrase NEAR}^-)' \text{hits}(' + ')}$$

where the hits() for all the elements of a set are added. For example:

$$\text{hits}(' + ') = \text{hits}(' excellent ') + \text{hits}(' good ') + \sum_{j=1,2,\dots,h} \text{hits}(\sigma_j)$$

with σ_j being represented by the synonymous terms which is added into the set from the WordNet during the process.

5 SMOOTHING RANKING WITH OPINION-BASED CLUSTERS

In this scheme we strive to use clustering information around the reviews to improve the ranking of entities. We use the algorithm ClustFuse of Kurland (Kurland, 2006), which makes use of two components to provide a score to a document d , the probability's relevance of the text to the query and the assumption that clusters can be used as proxies for the texts, that rewards texts belonging "strongly" in a cluster which is very relevant to the query.

The ClustFuse algorithm uses cluster information to improve the ranking. In summary, the algorithm in order to create a document ranking to a query q , creates a set of similar queries to q , let it be $Q = \{q_1, q_2, \dots, q_k\}$, and for each one of them receives a text ranking L_i . Then it tries to exploit clustering to all texts in the rankings L_i . Finally it produces a final

text ranking using the following equation:

$$p(q | d) = (1 - \lambda)p(d | q) + \lambda \sum_{c \in CL} p(c | q)p(d | c)$$

In our case as CL we set all review texts (r_i). To create the set of queries $Q = \{q_1, q_2, \dots, q_k\}$ for each query q , we use combinations of synonyms of the terms from the semantic network WordNet. We employ the Vector space model for representing review texts (r_i), the cosine similarity as texts distance metric, the k-means algorithm for clustering, the FcombSUM (d, q) as fusion method (Kurland, 2006), and the BM25 metric for assessing r_i to the questions and produce the ranked list L_i . Also as r_i 's features we use the sentiment ratings of aspects as they are obtained by the process described above in Section 4. So each cluster will consist of review texts with similar ratings in aspects. A detailed presentation of Kurland's scheme and an interpretation of the equations is presented in (Kurland, 2006).

6 THE NAIVE CONSUMER MODEL

In the previous schemes we employed a set of aspects keywords (features) as queries and evaluated the review texts on the relativity with those. But we consider that all aspects are equally important to be used in the assessment of the entities. For example, in the domain of the car we may say that the aspect "fuel consumption" is more important than the aspect "leather seats". It may not. We believe that the answer can only be given by the community. So we retrieve the appropriate information from the web; let D_{inf} be the set of those texts.

With this model we attempt to simulate the behavior of a consumer who is trying to assess entities from a specific domain and he knows some aspects, but he does not know the importance that each aspect has as criteria in the assessment. Usually such a user consults the web, for relevant articles in Wikipedia, in blogs, in forums, as well in sites that contain reviews of other users, to understand the importance that each aspect has.

We are attempting to collect the knowledge of D_{inf} , on which of the features are more important. We create a set of queries $Q = \{q_1, q_2, \dots, q_k\}$ containing the aspect query keywords. Each of the elements of Q is given as a query in a web search engine and the first ten results are being collected. Considering that search engines use a linear combination of measures such as BM25 and

PageRank (Page, Larry, 2002), (Baeza-Yates and Ribeiro-Neto, 2011), we can say that all the texts (pages) that we collect are relevant to the entities' domain which we examine, and that those texts are important nodes in the graph of the web, so important for the community.

Concerning the significance of a term t in a document d , as part of a text collection, we can say that is calculated from the BM25 score of the term t in d . Having the D_{inf} set of all texts, we are trying to extract how important is each aspect query keyword (a_i) for the entity domain that we examine, by calculating a score of significance. For this computation we can apply many formulas, but we choose to use the following which contains the participation rate of the feature a_i in the score of reviews:

$$scoreA(a_i) = \frac{\sum_{d \in D_{inf}} BM25(a_i)_d}{\sum_{a' \in A} (\sum_{d \in D_{inf}} BM25(a')_d)} \quad (1)$$

This formula tends to be similar with the Rasch model. In the analysis of data with a Rasch model, the aim is to measure each examinee's level of a latent trait (e.g., math ability, attitude toward capital punishment) that underlies his or her scores on items of a test. In our case the test is the assessment of the reviews, the examinees are the aspects, and the items of the test are the review texts.

Based on this idea we develop two models NCM1 and NCM2.

6.1 NCM1

Having the rate $scoreA(t)$ for each aspect, expressing how important this feature is, when used to evaluate entities of a particular class, we can combine it with the term that expresses how important each aspect for a specific review text (r_i) as part of a text collection (R), in order to assess reviews in queries consisting of aspect keywords, as follows:

$$p(q, r_i) = \sum_{t \in q} score(t)_{r_i} * scoreA(t) \quad (2)$$

where $p(q, r_i)$ is the probability of relevance between the query and the review r_i , $score(t)_{r_i}$ is the BM25 score of the word / aspect t for the review text r_i and $scoreA(t)$ is derived from (1).

6.2 NCM2

NCM2 works as at NCM1 applying additionally the Kurland's schema (Kurland, 2006), to exploit any

cluster organization, which may exist across the review texts. We employ equation (2) to assess the review texts to all queries $Q = \{q_1, q_2, \dots, q_k\}$ and create the L_i lists, where $\text{score}_A(t)$ is the importance score of aspects for their use in the evaluation process of entities, as it is calculated from the set of texts collected from the web. We use the algorithm ClustFuse as shown previously, in section 5.

7 EXPERIMENTS

We performed two sets of experiments to test the performance of our schemes, using two different datasets respectively. The datasets consist of sets of entities that are accompanied by users' reviews, which come from online sites. The queries consist of aspects keywords. For each one of the queries we produce the ideal entities' ranking based on the ratings given by the users in aspects together with the texts of the reviews. It is calculated as the average of the ratings given by each user for a certain characteristic as the Average Aspect Rating (AAR). For queries that are composed by several aspects, the average of the AAR aspects' scores of the question is calculated as the Multi-Aspect AAR (MAAR). More specifically, consider a query $q = \{a_1, a_2, \dots, a_m\}$, with m aspects as keywords, and an entity e , then $r_i(e)$ is the AAR of the entity e for the i -th aspect. Consequently MAAR is calculated as follows:

$$MAAR(e, q) = \frac{1}{m} = \sum_{i=1}^m r_i(e)$$

In the first set of experiments we use the OpinRank Dataset, which was presented in (Ganesan and ChengXiang, 2012) and consists of entities, which are accompanied by reviews of users from two different domains (cars and hotels). The reviews come from the sites Edmunds.com and Tripadvisor.com respectively. We use the reviews from the domain of the cars which includes car models and the corresponding reviews, for the years 2007-2009 (588) and we perform 300 queries. The texts of the reviews have averaged about 3000 words.

In the second set of experiments we use a collection of review texts for restaurants from the website www.we8there.com. Each review is accompanied by a set of 5 ratings, each in the range 1 to 5, one for each of the following five features {food, ambience, service, value, experience}. These scores were given by consumers who had written the

reviews. In the second set of experiments we use 420 texts with reviews, averaging 115 words, as published on the link: <http://people.csail.mit.edu/bsnyder/naacl07/data/> and we perform 31 queries. In this set of experiments, we also compare the performance of our schemas with a multiple aspect online ranking model which is presented in (Snyder and Barzilay, 2007), and is based on the algorithm Prank which is presented by Crammer and Singer in (Crammer and Singer, 2001). This supervised technique has shown that it delivers quite well in predicting the ratings on specific aspects of an entity using reviews of users for this. To create an m -aspect ranking model we use m independent Prank models, one for each aspect. Each of the m models, are trained to correctly predict one of the m aspects. Having represented the review texts r_i as a feature vector $x \in R^n$, this model predicts a score value $y \in \{1, \dots, k\}$ for each $x \in R^n$. The model is trained using the algorithm Prank (Perceptron Ranking algorithm), which reacts to incorrect predictions during training, updating the weight (w) and limits (b) vectors.

We evaluate the performance of the our schemas to produce the correct entity ranking, calculating the nDCG at the first 10 results.

7.1 Experimental Results

Initially we compare the performance of the BM25 model with and without the AvgScoreQAM and opinion expansion extensions, which are presented in (Ganesan and ChengXiang, 2012). We note that in both sets of experiments we conducted, using the BM25 model with the proposed extensions gives better results. This is one of the main observations in (Ganesan and ChengXiang, 2012), and here it is verified. In our measurements, however, we did not observe the expected increase in performance at the first set of experiments. It should be noted that in our experiments we did not use pseudo feedback mechanism, as in (Ganesan and ChengXiang, 2012).

The experimental results depict that the use of sentiment information present in reviews on the evaluation of the entities, can be used equally well as the conventional information retrieval techniques, such as the use of the BM25 metric. Both sentiment schemas perform sentiment analysis at sentence level, each in a different way. In the first set of experiments, our schemas show that they almost perform the same, while in the second set the technique using syntactic patterns evinces better than that using the sentiment lexicon. We believe that the better performance of the syntactic patterns-

based sentiment analysis in the second dataset is probably due to the fact that in the second dataset the reviews are small (average 115 words) and users express immediately and clearly their opinions forming simple expressions. It should be noted that although we chose simple unsupervised sentiment analysis techniques which do not perform in-depth analysis, we hoped that they would exceed in performance the information retrieval approach. This is because they have the ability to recognize and negative opinions, knowledge that ignores a model like BM25std+AvgScoreQAM+opinExp. However we do not observe this. We still believe that with more sophisticated sentiment techniques that can be accomplished. We must not forget that sentiment analysis techniques have to deal with the diversity of human expression. People use many ways to express their opinions, and there are many types of opinions. On the other hand, the performance and the simplicity of the information retrieval approach makes it an attractive option.

More we observe the performance of our two clustering models, the BM25std+Kurland and the BM25std+Kurland+opinion-based clusters, with which we seek to exploit clustering information from the review texts to improve the ranking of entities. In the first set of experiments BM25std+Kurland performs well, while in the second has low performance. The low performance of BM25std+Kurland is probably due to the fact that the texts in the second dataset are small in length (average words per text 115 words). So its representation in the vector space characteristics are similar, which introduces noise in the clustering process using the k-means algorithm. The BM25std+Kurland+opinion-based clusters scheme, performs well in both experiment sets. Also it is always better than that the standard BM25 formula and the BM25std+Kurland schema. Thus we can say that opinion based clustering can identify similar assessment behaviors to similar aspect queries among the entities and use this information to make a better entity ranking. This also shows that the opinion-based clustering is more suitable for an opinion-based entity ranking process than the content clustering.

Both of the naive consumer models show to perform better in the first set of experiments, while in the second set of experiments the schema that uses the Kurland technique and makes use of the cluster information, seems to overcome even the supervised classifier technique of the m-aspect prank model. So we see that it indeed plays an important role the knowledge of the importance of each

attribute used in the entities assessment, and also the knowledge of the aspects groups as they are defined by the users' community.

Table 1: We present the average of the nDCG@10 of the questions for all schemes on the two set of our experiments.

| method | 1 st exp. set | 2 nd exp. set |
|--|--------------------------|--------------------------|
| | | |
| BM25std | 0.87 | 0.936 |
| BM25std+AvgScoreQAM+opinExp | 0.88 | 0.955 |
| lexicon-based SA | 0.865 | 0.91 |
| syntactic patterns-based SA | 0.869 | 0.94 |
| BM25std+kurland | 0.88 | 0.90 |
| BM25std+Kurland+opinion-based clusters | 0.89 | 0.956 |
| NCM1 | 0.891 | 0.938 |
| NCM2 | 0.893 | 0.96 |
| m-aspect Prank | - | 0.95 |

8 CONCLUSIONS

In this paper we examined the problem of ranking entities. We developed schemes, which take into account sentiment and clustering information, and we also propose the naive consumer model. In order to supply more analytical hints we need more experiments for various application areas and this is a topic of future work however in this paper we aimed at providing a proof of concept of the validity of our approach. The information retrieval approach with the two extensions, the aspect modeling and the opinion expansion, presented in (Ganesan and ChengXiang, 2012), is a working and attractive option. The NCM model can be used to reveal more reliable entity rankings, thanks to the knowledge it extracts from the web. The opinion-based clustering schema can be also used to generate more accurate entity rankings. Regarding the sentiment analysis techniques, which are those that would probably give the complete solution on the entity ranking problem, for now, they are dependent on the level of analysis and on the characteristics of the opinionated text. The syntactic patterns based sentiment analysis technique in the second set of our experiments has better performance than the lexicon-based sentiment analysis. In the second dataset the reviews are small and users express immediately and clearly their opinions forming simple expressions, while in the first dataset the reviews are longer and opinion extraction becomes complex. Although there are

datasets that contain short texts, such as twitter datasets, in which opinion extraction can be quite difficult and require techniques that perform deeper sentiment analysis.

REFERENCES

- Amati, G., and van Rijsbergen, C. J., Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Trans. Inf. Syst.*, 20(4):357/389, 2002.
- Baeza-Yates, R., and Ribeiro-Neto, B. *Modern Information Retrieval: the concepts and technology behind search*. Addison Wesley, Essex, 2011.
- Crammer K., Singer Y., Pranking with ranking. *NIPS* 2001, 641-647.
- Fang H. and Zhai C., Probabilistic models for expert finding. *ECIR 2007*: 418-430.
- Fellbaum, C., editor. WordNet, an electronic lexical database. *The MIT Press*.1998.
- Gamon M., Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. *COLING* (2005), pp. 841-847 .
- Ganesan, K., and ChengXiang Z., Opinion-Based Entity Ranking. *Inf. Retr.* 15(2): 116-150 (2012).
- Hambleton, R. K., Swaminathan, H., and Rogers, H. J. *Fundamentals of Item Response Theory*. Newbury Park, CA: Sage Press (1991).
- He, Q., *Text Mining and IRT for Psychiatric and Psychological Assessment*. Ph.D. thesis, University of Twente, Enschede, the Netherlands. (2013).
- Kurland Oren, *Inter-Document similarities, language models, and ad-hoc information retrieval*. Ph.D. Thesis (2006).
- Liu Bing, *Opinion/Sentiment Lexicon* <http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html>.
- Liu Bing, *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers, 2012 .
- Nasukawa T. and Yi J., Sentiment analysis: capturing favorability using natural language processing. *Proceedings K-CAP '03 Proceedings of the 2nd international conference on Knowledge capture*, pp. 70-77 .
- Page, Larry, PageRank: Bringing Order to the Web. *Proceedings, Stanford Digital Library Project, talk. August 18, 1997* (archived 2002).
- Pang B. and Lee L., A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *Proceedings, ACL'04 Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*.
- Pang B. and Lee L., Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *ACL 2005*.
- Prabowo R. and Thelwall M., Sentiment analysis: A combined approach. *Journal of Informetrics, Volume 3, Issue 2, April 2009, Pages 143–157*.
- Rasch, G., Probabilistic Models for Some Intelligence and Attainment Tests, (Copenhagen, Danish Institute for Educational Research), with foreword and after word by B. D. Wright. The University of Chicago Press, Chicago (1960/1980).
- Robertson, S., Zaragoza, H., *The Probabilistic Relevance Framework: BM25 and Beyond.*, Foundations and Trends in Information Retrieval 3(4): 333-389 (2009).
- Snyder B. and Barzilay R., Multiple aspect ranking using the good grief algorithm. *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pp. 300-307.
- Tikves, S., Banerjee, S., Temkit, H., Gokalp, S., Davulcu, H., Sen, A., Corman, S., Woodward, M., Nair, S., Rohmaniyah, I., Amin,A., A system for ranking organizations using social scale analysis, *Soc. Netw. Anal. Min.*, (2012).
- Titov, Ivan and Ryan McDonald, A joint model of text and aspect ratings for sentiment summarization., *In Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-2008).*, (2008a).
- Titov, Ivan and Ryan McDonald, Modeling online reviews with multi-grain topic models., *In Proceedings of International Conference on World Wide Web (WWW-2008)*. 2008b. doi:10.1145/1367497.1367513.
- Tsuruoka Yoshimasa, *Lookahead POS Tagger*, <http://www.logos.t.u-tokyo.ac.jp/~tsuruoka/lapos/>.
- Turney, P.D, *Thumbs up or thumbs down? Semantic Orientation Applied to Unsupervised Classification of Reviews*. ACL, pages 417-424. (2002).
- Turney P. D. and Littman M. L., Measuring praise and criticism: Inference of semantic orientation from association. *In Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL 2002)*.
- Wang H., Lu Y., and Zhai C., Latent aspect rating analysis on review text data: a rating regression approach. *In Proceedings KDD '10 Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 783-792 (2010).
- Yu, Jianxing, Zheng-Jun Zha, Meng Wang, and Tat-Seng Chua, Aspect ranking: identifying important product aspects from online consumer reviews. *In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*. (2001).
- Zhai, C. and Lafferty, J., A study of smoothing methods for language models applied to ad hoc information retrieval. *In Proceedings of SIGIR'01*, pp. 334–342 (2001).
- Rasch, *The Rasch model*, http://en.wikipedia.org/wiki/Rasch_model.
- ITL, *Item Response Theory*, http://en.wikipedia.org/wiki/Item_response_theory.
- List of part-of-speech tags, http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html.

Handling Weighted Sequences Employing Inverted Files and Suffix Trees

Klev Diamanti¹, Andreas Kanavos², Christos Makris² and Thodoris Tokis²

¹Science for Life Laboratory, Department of Cell and Molecular Biology, Uppsala University, Sweden

²Department of Computer Engineering and Informatics, University of Patras, Greece
kdiamanti@outlook.com, kanavos, makri, tokis@ceid.upatras.gr

Keywords: Searching and Browsing, Web Information Filtering and Retrieval, Text Mining, Indexing Structures, Inverted Files, n-gram Indexing, Sequence Analysis and Assembly, Weighted Sequences, Weighted Suffix Trees.

Abstract: In this paper, we address the problem of handling weighted sequences. This is by taking advantage of the inverted files machinery and targeting text processing applications, where the involved documents cannot be separated into words (such as texts representing biological sequences) or word separation is difficult and involves extra linguistic knowledge (texts in Asian languages). Besides providing a handling of weighted sequences using n -grams, we also provide a study of constructing space efficient n -gram inverted indexes. The proposed techniques combine classic straightforward n -gram indexing, with the recently proposed two-level n -gram inverted file technique. The final outcomes are new data structures for n -gram indexing, which perform better in terms of space consumption than the existing ones. Our experimental results are encouraging and depict that these techniques can surely handle n -gram indexes more space efficiently than already existing methods.

1 INTRODUCTION

In this paper we focus on handling weighted sequences (Makris and Theodoridis, 2011). The difference between weighted sequences and regular strings is that in the former, we permit in each position the appearance of more than one character, each with a certain probability (Makris and Theodoridis, 2011). Specifically, a weighted word $w = w_1 w_2 \dots w_n$ is a sequence of positions, where each position w_i consists of a set of couples; each couple has the form $(s, \pi_i(s))$, where $\pi_i(s)$ is the probability of having the character s at position i . Also, for every position w_i , $1 \leq i \leq n$, $\sum \pi_i(s) = 1$. Moreover, it is usually assumed that a possible subword is worth the effort to be examined if the probability of its existence is larger than $1/k$; with k being a user defined parameter. In order to handle weighted sequences the *Weighted Suffix Tree* data structure was implemented (Iliopoulos et al., 2006). We consider this specific data structure as a proper suffix tree generalization.

The novelty in our approach is that for the first time, we exploit inverted files and n -grams in the handling of weighted sequences, thus providing an interesting alternative to weighted suffix trees for a variety of applications that involve weighted sequences. Our approach is interesting since it offers interesting al-

ternatives to approaches using suffix arrays and suffix trees with inverted files. This lacked in the bibliography in contrast to traditional pattern search applications such as in search engines where both alternatives were offered (see for example (Puglisi et al., 2006)). We do not delve into details of various pattern matching operations but merely focus on how to space efficiently transform weighted sequences into normal and then handle them using the well known technique of n -grams. Our target is not only at biological, but also at natural language applications. n -grams are sequences of consecutive text elements (either words or symbols); they are widely used in Information Retrieval (Ogawa and Iwasaki, 1995), (Lee and Ahn, 1996), (Navarro and Baeza-Yates, 1998), (Millar et al., 2000), (Navarro et al., 2000), (Navarro et al., 2001), (Gao et al., 2002), (Mayfield and McNamee, 2003), (Kim et al., 2007), (Yang et al., 2007), especially in applications employing text that cannot be separated into words.

The indexes produced with the n -gram inverted index technique, have a number of advantages. One of them is that they work on any kind of sequences, even if the sequence consists of words which have no practical meaning, such as DNA and protein sequences. Moreover, the n -gram technique is language neutral since it can be applied on different languages. Ano-

ther major benefit is that this indexing method is error-tolerant, putting up with errors that occur during the construction of the index; this is as it uses for its construction, the 1-sliding technique.

Nevertheless, the n -gram inverted index has also some drawbacks; the size tends to be very large and the performance of queries tends to be inefficient. This is the reason why a wide amount of research on how to use this technique space efficiently has been performed (Kim et al., 2005), (du Mouza et al., 2009), (Tang et al., 2009).

In (Kim et al., 2005), an efficient method for constructing a two-level index is proposed. Specifically, this method reduces significantly the size of the index and improves the query performance when comparing to the straightforward n -gram inverted index technique; while preserving all the advantages of the n -gram inverted index. This technique extracts substrings of fixed length m from the original sequence and then applies the classic n -gram technique on each of those extracted substrings. As shown in (Kim et al., 2005), this technique can provide significant space improvements, but as it can be observed in our experimental results, when the original sequence is not enough repetitive, the performance of this two-level indexing technique deteriorates.

In detail, we propose three new techniques for handling weighted sequences using n -grams indexing. We additionally propose a new framework for space compaction aiming to face the aforementioned space shortcomings of (Kim et al., 2005). In our space efficient framework, instead of resorting to the two-level indexing scheme, we judiciously select a set of substrings of the initial sequences for the n -grams of which, we employ the two-level indexing scheme; while for the rest of them, we employ the straightforward one-level indexing scheme. The substrings are selected based on the frequency of their appearance in the whole document set. Also, the length of substrings covering the initial sequence as well as the two distinct variants of the algorithmic scheme (variant for selecting these substrings employing a forest of suffix trees and a variant for the generalized suffix tree) are implemented and tested. It should be noted that these generalized suffix trees are the weighted suffix trees derived from the initial set of weighted sequences.

What is more, experiments on both synthetic and real data are performed in order to validate the performance of our constructions and the space reduction that they offer. Our work can be considered both an experimental research for the weighted sequences as well as a survey for validating the space efficiency of newly and previously proposed constructions in the area of n -gram indexing.

The rest of the paper is organized as follows. In section 2, the related work as well as the contribution is presented. In section 3, we present the techniques for handling weighted sequences. Subsequently, in section 4, we describe our space compaction heuristics. In following, section 5 presents a reference to our experimental results. Finally, section 6 concludes the paper and provides future steps and open problems.

2 RELATED WORK AND CONTRIBUTION

In (Christodoulakis et al., 2006), a set of efficient algorithms for string problems, involving weighted sequences arising in the computational biology area, were presented adapting traditional pattern matching techniques to the weighted scenario. What is more, in order to approximately match a pattern in a weighted sequence, a method was presented in (Amir et al., 2006) for the multiplicative model of probability estimation. In particular, two different definitions for the Hamming as well as for the edit distance, in weighted sequences, were given. Furthermore, we should refer to some more recent techniques (Zhang et al., 2010a), (Zhang et al., 2010b), (Alatabbi et al., 2012), that besides extending previous approaches, they also employ the Equivalence Class Tree for the problem at hand. From these papers, special mentioning deserves the work in (Zhang et al., 2010a), which generalizes the approach in (Iliopoulos et al., 2006), so as to handle effectively various approximate and exact pattern matching problems in weighted sequences.

In addition, there is a connection with the probabilistic suffix tree, which is basically a stochastic model that employs a suffix tree as its index structure. This connection aims to represent compactly the conditional distribution of probabilities for a set of sequences. Each node of the corresponding probabilistic suffix tree is associated with a probability vector that stores the probability distribution for the next symbol, given the label of the node as the preceding segment (Marsan and Sagot, 2000), (Sun et al., 2004).

In our work, we will mainly employ the preprocessing techniques presented in (Iliopoulos et al., 2006), where an efficient data structure for computing string regularities in weighted sequences was presented; this data structure is called *Weighted Suffix Tree*. Our approach however can be also modified to incorporate the techniques presented in (Zhang et al., 2010a).

The main motivation for handling weighted sequences comes from Computational Molecular Bio-

logy. However, there are possible applications in Cryptanalysis and musical texts (see for a discussion but in this time for the related area of Indeterminate Strings, which are strings having in positions, sets of symbols, (Holub and Smyth, 2003), (Holub et al., 2008)). In Cryptanalysis, undecoded symbols may be modeled as set of letters with several probabilities, while in music, single notes may match chords or notes with several probabilities. In addition, our representation of n -grams and our space compaction heuristics are of general nature concerning the efficient handling of multilingual documents in web search engines and general in information retrieval applications.

Character n -grams are used especially in CJK (Chinese, Japanese and Korean) languages, which by nature cannot be easily separated into words. In these languages, 2-gram indexing seems to work well. For example in (Manning et al., 2008), it is mentioned that in these languages, the characters are more like syllables than letters and that most words are small in numbers of characters; also, the word boundaries are small and in these cases, it is better to use n -grams. Moreover, n -grams are helpful in Optical Character Recognition where the text is difficult to comprehend and it is not possible to introduce word breaks. Additionally, k -grams are useful in applications such as wildcard queries and spelling correction.

3 ALGORITHMS

We initially describe the n -gram based techniques for handling normal sequences, which are being presented in (Kim et al., 2005). Then we explain how these can be adapted so that we can handle weighted sequences. The algorithm proposed in (Kim et al., 2005) tries to improve the straightforward inverted file scheme that produces n -grams on the fly using a sliding window; afterwards the algorithm stores them in an inverted file by replacing it with a two-level scheme, which is shown to be more space efficient.

In particular, this novel two-level scheme is based on the following approach: (i) each of the initial sequences is processed and a set of substrings of length m is extracted so as to overlap with each other by $n - 1$ symbols, (ii) an inverted index (called back-end index) for these substrings as well as the initial sequence set, considering the substrings as distinct words, are built, (iii) all the n -grams in each of the substrings are extracted, (iv) an inverted index (called front-index) is built, regarding the substrings as documents and the n -grams as words. This scheme, called by its authors n -gram/2L, can be applied to any text

and in some cases, results to significant space reduction.

If the text can be partitioned into words (natural language text), another scheme termed n -gram/2L-v is provided. So, the subsequences are defined as consecutive sequences of the text words, by exploiting the intuitive remark that words exhibit repetitiveness in natural language text. Their experiments show that when applied to natural text n -gram/2L-v, sample space savings, compared to the initial technique, are produced.

We attempt to adapt their techniques by presenting three algorithms for handling weighted sequences, which are based in the exploitation of the technique presented in (Kim et al., 2005); then we can adjust them to the problem at hand.

3.1 1st Technique - Subsequences Identification

In the first technique, we form separate sequences as we split each weighted sequence into weighted substrings; each one of length m . Each one of these weighted substrings is used to produce normal substrings by employing the normal substrings generation phase of (Iliopoulos et al., 2006) (p.267, algorithm 2). In this phase, the generation of a substring stops when its cumulative possibility has reached the $1/k$ threshold. The cumulative possibility is calculated by multiplying the relative probabilities of appearance of each character in every position. Each produced substring is of maximum size m and for every substring, we produce all the possible n -grams. After this procedure, we store all the produced n -grams in the n -gram/2L-v scheme.

Concerning the generation phase, all the positions in the weighted sequences are thoroughly scanned and at each branching position, a list of possible substrings, starting from this position, is created. Then moving from left to right, the current subwords are extended by adding the same single character whenever a non-branching position is encountered; in contrast there is also a creation of new subwords at branching positions where potentially many choices are supplied.

3.2 2nd Technique - On the fly n -grams Identification

This technique is much simpler as we don't need to deploy all the generic sequences. Unlike the previous technique, we just need to produce all the possible n -grams and in following for each report, its corresponding weighted sequences as well as their offsets.

As a matter of fact, we don't have to form separate sequences, as in the previous approach, but instead only split each generalized sequence into segments, each of size m , and for each segment, just produce the requested n -grams.

Hence, this particular scheme is by nature one-level and we propose its use due its simplicity. However, as it will be highlighted in the experiments, there are cases when the technique outperforms the previous one in terms of space complexity.

4 SPACE EFFICIENT INVERTED FILE IMPLEMENTATIONS FOR NORMAL SEQUENCES

Our crucial remark is that, in order for the n -gram/2L technique to provide space savings, the substrings, where the initial sequences are separated, should appear a large number of times and should cover a broad extent of the initial sequences, otherwise in case this does not apply (e.g. if there is a large number of unique substrings), then the space occupancy turns out to increase instead of shrinking.

Hence, it would be preferable to use a hybrid scheme instead of a two-level one; there we should extract from the initial sequences, substrings that appear repetitively enough and cover a large extent of the initial sequences. In following, for the specific substrings, we will employ a two-level scheme; while for the remaining parts of the sequences, we will use the straightforward one-level representation. During this separation, we elongate each selected substring by $n-1$, as in (Kim et al., 2005).

So, as to achieve our goal and build a hybrid one and two-level inverted index, we introduce three techniques:

4.1 One Simple Technique

A variant of the algorithm described in (Kim et al., 2005), called Hybrid indexing Algorithm version 0 - hybrid(0), is implemented. In this implementation, we decided to store the substrings of length m and of a number of occurrences in the back-end inverted file of the two-level scheme; provided that this number is greater than a trigger. The user is asked to provide the value of the trigger; the trigger is set equal to 1, for the results presented in the corresponding section.

The substrings, occurring less or equal to the provided trigger, are just decomposed in their n -grams and then saved in a one-level index. The substrings stored in the two-level scheme, are also decomposed

in their n -grams, which we forward to the front-end index of the two-level scheme.

4.2 Two Techniques based on Suffix Trees

In these techniques, we locate substrings that (in contrast to hybrid(0)) can be of varying size, highly repetitive and cover a large extent of the initial sequences. So as to locate them, we employ suffix trees (McCraith, 1976) that have been previously used in similar problems (Gusfield, 1997) of locating frequent substrings. In particular, we provide two different variants in the implementation of our space efficient heuristic schema. Those two distinct versions share a common initial phase, while differing in their subsequent workings.

More analytically, we insert all the sequences in a generalized suffix tree as described in (Gusfield, 1997) and in following we use this tree for counting the repetitions of each substring of the stored documents. Note that if the sequences have been produced by using mappings from weighted sequences, then the produced suffix tree is similar to the weighted suffix tree of the initial sequences. This operation is performed during the building of the generalized suffix tree; after that, each node of the tree keeps the information concerning the repetitions of the substrings stored in it.

Subsequently, in each repetition, our algorithm chooses a substring and a subset of each occurrence. These two objects are in following included in the two-level index. The selection procedure is described as follows:

1. The substring needs to have a length equal or greater than s ; s is the least acceptable length of a substring and constitutes a user defined parameter at the start of the algorithm's execution.
2. The substring has to be highly repetitive. This means that it should have more than a specific number of occurrences (trigger) in the set of indexed documents; this trigger is also a user defined parameter.
3. The appearances of the selected substring, which are to be included in the two-level index, should not overlap in more than half the length of the subsequence; i.e. if the substring has a length of 10 characters, consecutive appearances of this substring should not overlap on more than 5 characters. By setting this criterion, we keep only the discrete appearances of the selected substring.

After the end of the procedure, we have selected a collection of substrings. We then sort this collection

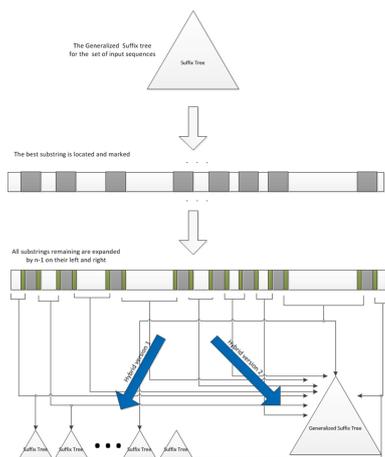


Figure 1: Visualizing hybrid(1) and hybrid(2) techniques.

based on the total length of the original sequences that the distinct occurrences cover (according to criterion 3). Furthermore, we select as best the occurrences of specific subsequence that cover the majority of the length of the initial sequences. We extract all these substrings from the initial sequences, thus including them in the two-level index. As a result, we have split the initial sequences into a set of partitions that are not included in the two-level index. Next, we elongate them by $n - 1$, so as not to miss any n -gram; where n is the n -gram length. Finally, we keep all these elongated substrings in a list. As a result, we have performed the preprocessing step that allows us to follow one out of two methods described below (see the procedure in Fig. 1):

(i) Hybrid Indexing Algorithm version 1 - hybrid(1). We construct for each elongated substring, a separate suffix tree and process best utilizing the same method as above. Then, our algorithm continues executing the process for each suffix tree constructed as cited above. This process is repeated as many times as the user chooses at the beginning of the algorithm execution.

(ii) Hybrid Indexing Algorithm version 2 - hybrid(2). We include all elongated substrings mentioned in a unified generalized suffix tree. In following, our algorithm executes the process for the generalized suffix tree constructed. This process is repeated as many times as requested. Generally, the more recursions we made, the better results we had; however, because of the limited system resources, we opted for 50 recursions in our experiments.

5 EXPERIMENTS

5.1 Experimental Setting

In our experiments, we used random weighted sequences to test our n -gram mapping techniques as well as one file (of size 1 GB) containing Protein data and DNA data to test our space compaction heuristics. We also performed experiments with 10MB and 100MB with similar results. Due to lack of space, only figures and comments from the 1GB data are presented in the main body of the article. Our experimental data were downloaded from the NCBI databases (<ftp://ftp.ncbi.nih.gov/genomes/>). Furthermore, we use initials to designate both m (length of substrings) as well as the parameter s (size in bytes) in our space compaction heuristics.

The computer system, where the experiments were performed, was an Intel Core i5-2410M 2.3 GHz CPU with a 3GB (1x1GB and 1x2GB in 2xDual Channel) RAM. The techniques we implemented and applied on the experimental data mentioned above, were:

1. Weighted Sequences Identification:
 - (i) Subsequences Identification,
 - (ii) On the fly n -grams Identification and
 - (iii) Offline Identification.
2. Space compaction heuristics:
 - (i) One-Level Inverted File (using the classic straightforward technique),
 - (ii) Two-Level Inverted File (using the technique in (Kim et al., 2005)),
 - (iii) Hybrid Inverted File using the Simple Technique - hybrid(0),
 - (iv) Hybrid Inverted File with separate suffix trees - hybrid(1) and
 - (v) Hybrid Inverted File with a unified generalized suffix tree - hybrid(2).

For our space compaction heuristics, we run all techniques proposed in this paper (hybrid(0), hybrid(1) and hybrid(2)) in order to identify the most space efficient solution available. So as to depict the space compaction effectiveness of our approach, we tried our approach on real data of significant size and performed several experiments. As the experiments show, our approach outstandingly reduces the space complexity and stands by itself as a considerable improvement.

5.2 Weighted Sequences Results

As is depicted in Fig. 2, the offline approach is the worst in the attained space complexity, as expected.

The reason is because all possible combinations of sequences are produced; not only those that are needed by the two-level scheme. On the other hand, the offline approach is more flexible since it can incorporate different values of variables n and s .

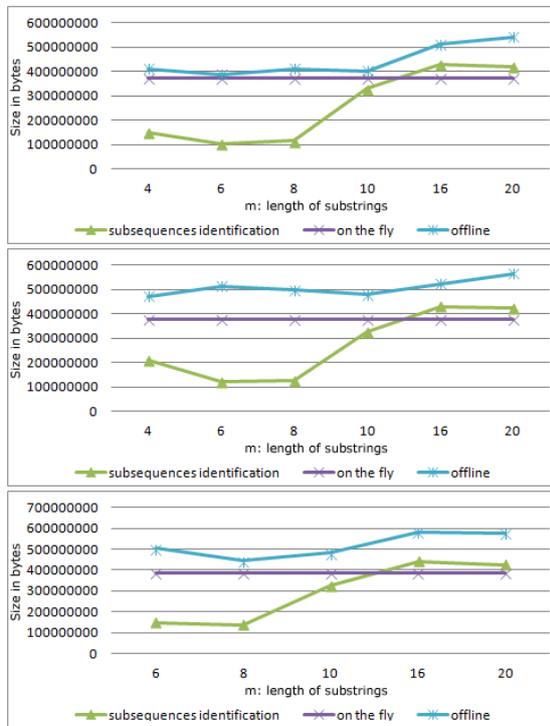


Figure 2: Weighted Sequences 10MB for varying size of s (a) $n=2$, (b) $n=3$ and (c) $n=4$.

With regards to the other two techniques, the on the fly approach is the most robust and stable in performance due to its fixed algorithmic behavior when handling every possible input. The identification of the subsequences, although better for small values of s , behaves worse for larger values. This can be attributed to the shortage of repetitions; being a vital ingredient of the success of this method's heuristic, when the value of s is increasing.

5.3 Protein Data Results

In the performed experiments, we never needed to make more than 50 recursions, as by this number we got the best possible results from the index method. Moreover, we ran experiments of substrings that have length from 4 to 10, in order to demonstrate the improvements that the two-level technique produces to the inverted file size.

Our hybrid(2) technique seems to be not as efficient as hybrid(1) is. Although, it theoretically considers the high repetitive sequence more efficiently

than the hybrid(1) technique, it does not seem to have satisfactory results. A probable explanation could be that using separate suffix trees, this method permits more choices in the sequences that will be selected for separate indexing than the Generalized suffix tree; the latter demands the selection of the same substring across different substrings. Furthermore, the technique is sensitive to the number of performed recursions and needs a vast number of them to work effectively.

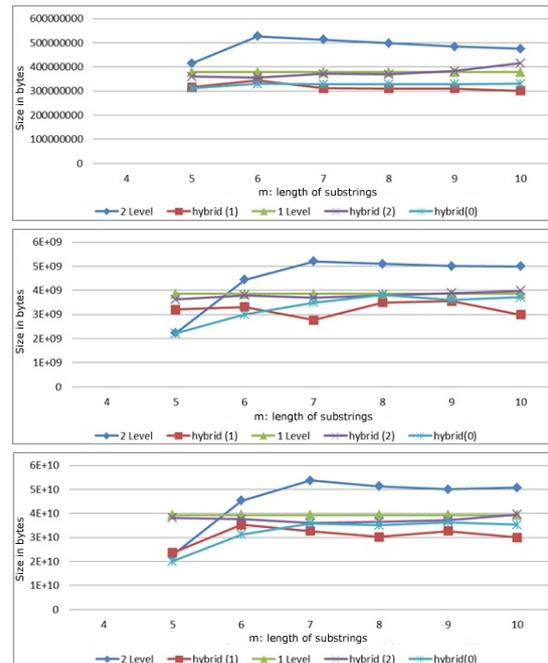


Figure 3: Protein Data 1GB for varying size of s (a) $n=2$, (b) $n=3$ and (c) $n=4$.

Another finding is that hybrid(0) technique is quite similar to the two-level technique for substrings with length 4 and 5 and after that, it is not as efficient as our hybrid(1) technique. This behavior can be explained from the fact that this technique always takes advantage of the positive characteristics of the two-level techniques as long as it is better than one-level; otherwise it resorts to the one-level.

Generally, in Protein data, our methods achieve better results due to the fact that they take advantage of the repetitiveness of the initial sequence even when the number of the repetitions is quite low. This is something that does not hold for the two-level scheme, where the performance is clearly degraded.

5.4 DNA Data Results

In the results shown below, the maximum number of recursions made, was fixed to 50 for each experiment.

In case of DNA data, we experimented for substrings that have length from 4 to 13. We examined more substring sizes so as to clarify the inefficiency of the two-level technique when the repetitiveness becomes lower. It is obvious that the two-level technique increases the inverted file size produced, when the substring length becomes larger than 11.

Analyzing the results presented in figure with DNA data results, we can patently see that our hybrid(1) technique is not as efficient as the two-level index. The reason for this inefficiency is that two-level index takes advantage of the substrings of length from 6 to 11, which seems to be highly repetitive in the DNA sequences examined. As soon as the size of the substring becomes lower than 6 or larger than 11, our method becomes obviously better. This occurs because the DNA data file used, is not so highly repetitive for subsequences of length <6 or >11 .

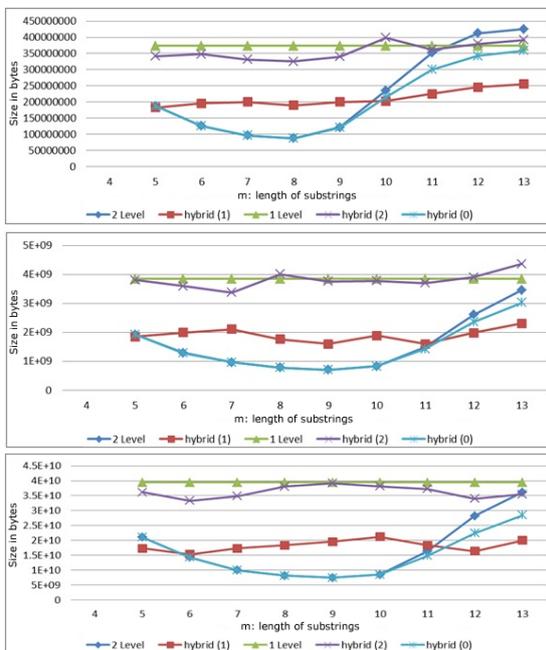


Figure 4: DNA Data 1GB for varying size of s (a) $n=2$, (b) $n=3$ and (c) $n=4$.

In cases when two-level technique performs better than hybrid(1), we use hybrid(0) to store our data. Hybrid(0) performs very similarly to two-level technique. The differences between the files produced by those two techniques are considered to be negligible. The reason why this phenomenon appears is due to the highly repetitive nature of DNA data (the limited alphabet) on limited size sequences.

As for our hybrid(2) method, we can clearly see that this method seems to be inefficient, and works worse than hybrid(1); this was something that was also noted in Protein data and can be explained in a

similar way as previously mentioned. Perhaps a better tuning of the involved algorithmic parameters and a combination with hybrid(1) would result in a more efficient scheme; but this is left as future work.

By choosing hybrid(0) or hybrid(1) techniques to save the DNA data in inverted indexes, we are led to very compact inverted file sizes. These sizes generally outperform or at least approximate the two-level index efficacy.

In conclusion, our experiments clearly prove that our techniques can significantly reduce space complexity by handling n -gram indexes and can also stand as considerable improvements.

6 GENERAL CONCLUSIONS AND FUTURE WORK

In this article we presented a set of algorithmic techniques for efficiently handling weighted sequences by using inverted files. Also, these methods deal effectively with weighted sequences using the n -gram machinery. Three techniques, which act as alternatives to other techniques that mainly use suffix trees, were presented. We furthermore completed our discussion by presenting a general framework that can be employed so as to reduce the space complexity of the two-level inverted files for n -grams.

In the future, we intend to experiment with various inverted file intersection algorithms (Culpepper and Moffat, 2010), in order to test the time efficiency of our scheme when handling such queries. We could perhaps incorporate some extra data structures as those in (Kaporis et al., 2003) as a well thought out plan. Last but not least, we also plan to apply our technique to natural language texts.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thales. Investing in knowledge society through the European Social Fund.

REFERENCES

Alatabbi, A., Crochemore, M., Iliopoulos, C. S., and Okanlawon, T. A. (2012). Overlapping repetitions

- in weighted sequence. In *International Information Technology Conference (CUBE)*, pp. 435-440.
- Amir, A., Iliopoulos, C. S., Kapah, O., and Porat, E. (2006). Approximate matching in weighted sequences. In *Combinatorial Pattern Matching (CPM)*, pp. 365-376.
- Christodoulakis, M., Iliopoulos, C. S., Mouchard, L., Perdikuri, K., Tsakalidis, A. K., and Tsihlias, K. (2006). Computation of repetitions and regularities of biologically weighted sequences. In *Journal of Computational Biology (JCB)*, Volume 13, pp. 1214-1231.
- Culpepper, J. S. and Moffat, A. (2010). Efficient set intersection for inverted indexing. In *ACM Transactions on Information Systems (TOIS)*, Volume 29, Article 1.
- du Mouza, C., Litwin, W., Rigaux, P., and Schwarz, T. J. E. (2009). As-index: a structure for string search using n-grams and algebraic signatures. In *ACM Conference on Information and Knowledge Management (CIKM)*, pp. 295-304.
- Gao, J., Goodman, J., Li, M., and Lee, K.-F. (2002). Efficient set intersection for inverted indexing. In *ACM Transactions on Asian Language Information Processing*, Volume 1, Number 1, pp. 3-33.
- Gusfield, D. (1997). *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*. Cambridge University Press.
- Holub, J. and Smyth, W. F. (2003). Algorithms on indeterminate strings. In *Australasian Workshop on Combinatorial Algorithms*.
- Holub, J., Smyth, W. F., and Wang, S. (2008). Fast pattern-matching on indeterminate strings. In *Journal of Discrete Algorithms*, Volume 6, pp. 37-50.
- Iliopoulos, C. S., Makris, C., Panagis, Y., Perdikuri, K., Theodoridis, E., and Tsakalidis, A. K. (2006). The weighted suffix tree: An efficient data structure for handling molecular weighted sequences and its applications. In *Fundamenta Informaticae (FUIN)*, Volume 71, pp. 259-277.
- Kaporis, A. C., Makris, C., Sioutas, S., Tsakalidis, A. K., Tsihlias, K., and Zaroliagis, C. D. (2003). Improved bounds for finger search on a ram. In *ESA*, Volume 2832, pp. 325-336.
- Kim, M.-S., Whang, K.-Y., and Lee, J.-G. (2007). n-gram/2l-approximation: a two-level n-gram inverted index structure for approximate string matching. In *Computer Systems: Science and Engineering*, Volume 22, Number 6.
- Kim, M.-S., Whang, K.-Y., Lee, J.-G., and Lee, M.-J. (2005). n-gram/2l: A space and time efficient two-level n-gram inverted index structure. In *International Conference on Very Large Databases (VLDB)*, pp. 325-336.
- Lee, J. H. and Ahn, J. S. (1996). Using n-grams for korean text retrieval. In *ACM SIGIR*, pp. 216-224.
- Makris, C. and Theodoridis, E. (2011). *Algorithms in Computational Molecular Biology: Techniques, Approaches and Applications*. Wiley Series in Bioinformatics.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
- Marsan, L. and Sagot, M.-F. (2000). Extracting structured motifs using a suffix tree - algorithms and application to promoter consensus identification. In *International Conference on Research in Computational Molecular Biology (RECOMB)*, pp. 210-219.
- Mayfield, J. and McNamee, P. (2003). Single n-gram stemming. In *ACM SIGIR*, pp. 415-416.
- McCreight, E. M. (1976). A space-economical suffix tree construction algorithm. In *Journal of the ACM (JACM)*, Volume 23, pp. 262-272.
- Millar, E., Shen, D., Liu, J., and Nicholas, C. K. (2000). Performance and scalability of a large-scale n-gram based information retrieval system. In *Journal of Digital Information*, Volume 1, Number 5.
- Navarro, G. and Baeza-Yates, R. A. (1998). A practical q-gram index for text retrieval allowing errors. In *CLEI Electronic Journal*, Volume 1, Number 2.
- Navarro, G., Baeza-Yates, R. A., Sutinen, E., and Tarhio, J. (2001). Indexing methods for approximate string matching. In *IEEE Data Engineering Bulletin*, Volume 24, Number 4, pp. 19-27.
- Navarro, G., Sutinen, E., Tanninen, J., and Tarhio, J. (2000). Indexing text with approximate q-grams. In *Combinatorial Pattern Matching (CPM)*, pp. 350-363.
- Ogawa, Y. and Iwasaki, M. (1995). A new character-based indexing organization using frequency data for japanese documents. In *ACM SIGIR*, pp. 121-129.
- Puglisi, S. J., Smyth, W. F., and Turpin, A. (2006). Inverted files versus suffix arrays for locating patterns in primary memory. In *String Processing and Information Retrieval (SPIRE)*, pp. 122-133.
- Sun, Z., Yang, J., and Deogun, J. S. (2004). Misae: A new approach for regulatory motif extraction. In *Computational Systems Bioinformatics Conference (CSB)*, pp. 173-181.
- Tang, N., Sidirourgos, L., and Boncz, P. A. (2009). Space-economical partial gram indices for exact substring matching. In *ACM Conference on Information and Knowledge Management (CIKM)*, pp. 285-294.
- Yang, S., Zhu, H., Apostoli, A., and Cao, P. (2007). N-gram statistics in english and chinese: Similarities and differences. In *International Conference on Semantic Computing (ICSC)*, pp. 454-460.
- Zhang, H., Guo, Q., and Iliopoulos, C. S. (2010a). An algorithmic framework for motif discovery problems in weighted sequences. In *International Conference on Algorithms and Complexity (CIAC)*, pp. 335-346.
- Zhang, H., Guo, Q., and Iliopoulos, C. S. (2010b). Varieties of regularities in weighted sequences. In *Algorithmic Aspects in Information and Management (AAIM)*, pp. 271-280.

XML Approximate Semantic Query based on Ontology

Yunkai Zhu, Chunhong Zhang and Yang Ji

Mobile Life and New Media Laboratory, Beijing University of Post and Telecommunications (BUPT), Beijing, China
ykzhu@mail.bnu.edu.cn, {zhangch.bupt.001, ji.yang.0001}@gmail.com

Keywords: Query Relaxation, Domain Ontology, NLP, RDF Triple, WordNet, SPARQL.

Abstract: More and more data is generated in XML format. How to effectively retrieve information from these data has attracted much research interest. Users have been used to keyword query without knowledge of data in advance. But XML data has additional structure than keywords. Almost all previous XML keyword queries require that user should be fully familiar with the XML structures and query syntax, which is not user-friendly and is seriously impediment to the prevalence of XML. In this paper, we propose to use natural sentence as query input because it can contain both keywords and their structure information. Query processing engine depends on NLP (Natural Language Process) technology and predefined templates to catch the query goal of user, optimally expressed as RDF (Resource Description Framework) triples. We exploit hierarchical structure relaxation based on query tree variation and vocabulary relaxation based on WordNet to relax input query. To better reflect the semantics of the query, we also use a certain domain OWL ontology constructed from XML schemas for reasoning and searching. Ontology gives us a reliable group of concepts and relations between the concepts. Ontology accurately transfers semantic information between human users and the computers. Finally we translate the RDF triples to SPARQL query sentences to retrieve RDF data.

1 INTRODUCTION

As XML data are generated more and more commonly in the Web and scientific applications, how to effectively retrieve information from these data has attracted much research interest (Guerrini, 2013). Keyword search has achieved great success on the web due to its merit of user-friendliness. As XML documents are more complex in structure, effective keyword search on XML documents requires deep understanding and special treatment of the structures in XML documents. Recently, a number of methods for XML keyword search have been proposed (Li, 2004; Cohen, 2003; Li, 2007), these methods pay considerable attention to the structural features of XML documents, and have shown their effectiveness. However, we find that almost all these methods ignore an important aspect of XML input query, which is the structure of the queries themselves. These methods require that user should be fully familiar with the XML structures and query syntax such as Xpath (<http://www.w3.org/TR/xpath/>) and XQuery (<http://www.w3.org/TR/xquery/>), which are W3C standards. (Liu, 2013) proposes keyword query with structure (QWS), but its recommended query unit

such as $q_{database}^{journal}$ can only accommodate a layer of relationship and is also not user-friendly. We need an XML search engine that has user-friendliness and accuracy.

XML data on the Web is characterized by data heterogeneity. Heterogeneity in XML data reflects the different value representation formats or structures. Heterogeneity may appear in the vocabulary and hierarchical structure: different tags may be employed in different collections to label the same information; the hierarchical structure of same documents in different sources may be slightly different. In order to cope with the heterogeneity of XML, we need to relax query. Query relaxation contains two aspects, vocabulary and hierarchical structure. As to vocabulary relaxation, specialists usually build XML tags and schemas, and users don't always share or understand their viewpoints. Users might not use the right keyword, leading to miss answers when writing a query. For example, a user might use mailbox instead of email. Users might also use an instance of some concept instead of concept itself. For example, a user may use *location* instead of *space*. As far as hierarchical structure relaxation, the query structure may not

correspond to the xml data structure, such as excess or lack of an intermediate node, or attribute nodes dislocation. So the hierarchical structure relaxation is indispensable in query processing. Consequently, approximate query processing is of importance for XML search.

Although conventional keyword search neglects the relationship between keywords, we can use a sentence instead of only keywords, in which we can accommodate both keywords and their relations. Querix and its query language allowing full English questions with a limited set of sentence beginnings is judged to be the most useful and best-liked query interface (Kaufmann, 2010). For higher accuracy and practicability of our system, we do some improvements in the sentence beginnings above Querix and form some rules of RDF triples abstractions. We use WordNet and domain ontology to verify and extend query in aspects of both vocabulary and hierarchical structure. At the same time, we note the credits of new extensive RDF triples, and show the result retrieved with extensive query descending by the credits.

As (Decker, 2000) points out, XML and RDF are the current standards for establishing semantic interoperability on the Web, but XML addresses only document structure. RDF better facilitates interoperation because it provides a data model that can be extended to address sophisticated ontology representation techniques. XTR-RTO (Xu, 2007) provides an approach to build OWL ontology using XML document. The construction approach firstly maps source of XML schema into RDF and then into OWL ontology.

Through a series of query processing like statements component analysis, query classification, and RDF triple abstraction, we finally transfer user query input to SPARQL (<http://www.w3.org/TR/rdf-sparql-query>) query language to retrieve RDF data.

To achieve query relaxation and take the advantages of keyword query and twig query, we propose our query system. Through natural language into RDF triples and triples expansion, we can connect user's demand with XML data. RDF triples can cover almost all the information of a query input. Triples expansion can adapt the user's query demand to XML data. Our contributions can be summarized as follows: 1) we propose connection general ontology WordNet with certain domain ontology to support semantic search. 2) We propose a user-friendly XML search interface in which user needn't know the structure of XML in advance. The user only needs to express his/her query requirements in the search box. 3) We introduce a

novel semantic ranking scheme to compute the relationship of expansive RDF triples with original ones. 4) We build a real system to support our semantic XML search architecture, in which we needn't worry about the difference of expression.

The rest of this paper is organized as follows: In Section 2, we review the related work. The architecture of our query system and several components are presented in Section 3. In Section 4, we present semantic similarity computation, followed by Section 5 where we do experiment on IOT (Internet of things) data. We conclude this paper in Section 6, where we provide a discussion on the overall agenda of ontology-based XML query relaxation and give the future ideas.

2 RELATED WORK

2.1 Keyword Query on XML

Various approaches have been proposed to identify relevant keyword matches. We divide these papers into three categories: LCA (lowest common ancestor) based approaches, which connect keyword matches and identify relevant matches using variants of LCA; statistics-based approaches, which identify relevant matches according to the statistics of the data; minimal tree/graph based approaches, which consider keyword matches in subtree/subgraph of the data that satisfy certain conditions as relevant.

A number of LCA based approaches have been proposed, including XSearch (Cohen, 2003), MLCA (Li, 2004), SLCA (Xu, 2003), and MaxMatch (Liu, 2008), etc. But these structures do not consider the semantic meanings of keywords. Among all the methods, only XSearch provides a ranking scheme. On the ranking scheme, (Golenberg, 2008) proposes to rank query results according to the distance between different keyword matches in a document. XSearch combines a simple TF/IDF IR ranking with tree size to rank results. However, its keyword input format requires users to have some knowledge of underlying schema information. This drawback limits its population. As a method based on statistics, XReal (Bao, 2010) exploits the statistics of underlying XML data to identify relevant matches and ranking. As the LCA concept does not apply for graph shaped XML documents (such as XML with ID/IDREF and RDF documents), the approaches that search XML graphs define a query result as a minimal subtree/subgraph of the XML graph that contains all or part of the query keywords. A minimal tree/graph is a subtree/subgraph of the data

graph such that no nodes can be removed from it and it is still connected and contains all query keywords. Both statistics-based approaches and tree/graph based approaches don't take the structure among keywords into consideration.

2.2 Twig Query

A twig query is a rooted labelled tree with two types of edges: /, child edge; and//, descendant edge. A child edge represents a downward edge in the XML document tree whereas a descendant edge represents a downward path. Twig pattern matching is essential in evaluating XPath/XQuery queries. There are four different approaches based on twig query: TopX (Theobald, 2008), Twig-Path Scoring (Amer-Yahia, 2005), TASM (Augsten, 2010), ArHeX (Sanz, 2008). All selected approaches support variations of twig queries, that is, tree patterns in which nodes represent the terms of which the user is interested in content part of the query and edges represent the structural relationships that the user wants to hold among the terms in structural part of the query. All these approaches require structural pattern input, and consider not enough to the vocabulary relaxation. We can surmise that the precision and recall of these methods are considerable. But this also require user to be fully familiar with the XML structures and query syntax.

3 SYSTEM ARCHITECTURE

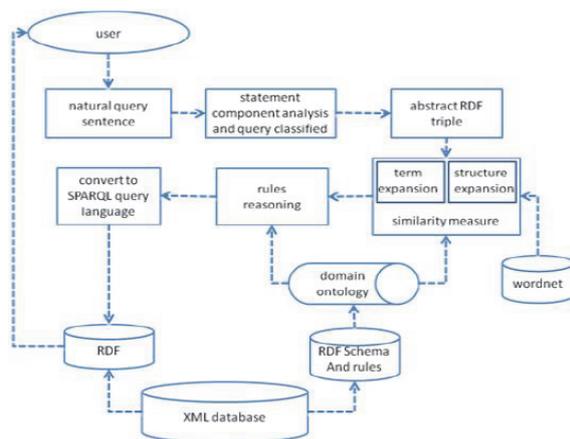


Figure 1: The architecture of our query system.

At a coarse-grained level of abstraction, our query system architecture can be characterized as a waterfall model, during which a NL (natural language) query is translated into a set of

intermediate and triple-based query triples, as is shown in figure 1. The main reason for adopting a triple-based data model is that it is possible to represent most queries as triples as <subject, predicate, object>. Based on the WordNet and domain ontology which is constructed from RDF schema and rules, triples are expanded to ontology-compatible triples, at the same time we measured the correlation of expansive triple to the original triples. We modified the triples according to the rules reasoning. Finally we convert the query triples to SPARQL query language, and then query the RDF abstracted from XML database, and finally feedback the related segments.

3.1 RDF Triples Abstraction from NL

The RDF is a framework for representing information in the Web. RDF triples can completely contain information in a sentence. An RDF triple contains three components: subject, predicate and object.

An RDF triple is conventionally written in the order subject, predicate, object. The predicate is also known as the property of the triple. A normal query sentence consists of key component such as subject, predicate, object and Modified ingredients (attributive, adverbial, complement, etc.).

3.1.1 Query Classification

All inquiries are sentences with question words which the user wants to get from the query. The type of a query has a great relationship with question words. This article proposes question classification based primarily on the correlation between question word and questions type. Its main purpose is to facilitate computer processing, and to help compute determine the question type. Through analyzing the types of questions, it is possible to catch the query intension of the user.

Based on the question word, questions can be classified into different categories. Some question words with a variety of questions tendency can't be directly determined, and have to be judged according to the wh-words collocations and semantic relations. Through segmentation and marking of common questions, we conclude questions classification mapping. After classification, the questions can be more precisely classified into predefined types. Its part classification is shown in Table 1.

Table 1: Wh-word classification.

| Question type | quantity | location | define | Thing | Person | Time |
|----------------|----------------------|----------|--------|-------|--------------|------------------------------|
| Question words | How many How much | Where | Why | What | Who Whose | Which year/month/day when |
| Mapping tag | quantity | location | reason | thing | Person | time |

3.1.2 Query Sentence Components and RDF Triples Mapping

In order to abstract the RDF triple of the query sentence, we need to analyse the structure of the sentence. The Stanford typed dependencies representation (<http://nlp.stanford.edu/>) is designed to provide a simple description of the grammatical relationships in a sentence that can be easily understood and effectively used by people who want to extract textual relations without linguistic expertise. In particular, it represents all sentence relationships uniformly as typed dependency relations, rather than the phrase structure representations that have long dominated in the computational linguistic community (De Marneffe, 2008). Through the dependencies analyzed by Stanford parser, we can know the synaptic structure of the query sentence and abstract the main information.

Here is an example sentence:

Who is manufacturer of temperature sensor which locates in beijing?

For this sentence, the SD (Stanford Dependencies) representation is a list of dependencies relations such as *nsub(is-2, manufacture-3)*, in which *nsub* representing the third word *manufancure* is subject of the second word *is*. These dependencies accompanying with sentence structure can map the query straightforwardly onto a directed graph representation, in which words in the sentence are nodes in the graph and grammatical relations are edge labels. Figure 2 gives the graph representation for the example sentence above.

Combining with XML features, we summarize some general rules (see in appendix) and templates of abstracting RDFs <subject, predicate, object> by ourselves. The rules in appendix give the details. As the figure above show, the red mark is original subject, but converts to predicate because of its following *Prep_of* structure. When the subject followed with *Prep_of* structure, the nouns after the *Prep_of* structure become the subject, and the nouns before degenerate to predicate. The green represents that it is a subsidiary part to the related

part. The blue represents object and the orange represents subject in both the RDF triple <temperature sensor, manufacturer, who> and <temperature sensor, locates, beijing>.

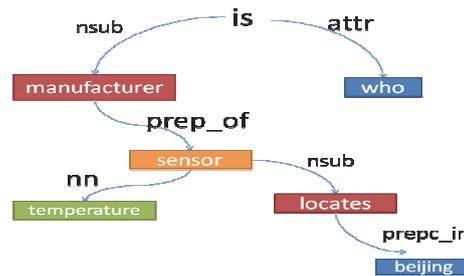


Figure 2: The grammatical tree.

3.2 RDF Triples Expansion

To prevent that the query result is empty and get more relevant results, we need to expand query. RDF triple expansion contains two aspects, vocabulary expansion and structure expansion. Triples abstracted from query sentence as <subject, predicate, object> can be translated into a query tree. The query tree is constructed by merging the same nodes which have the same words and location index. We first relax the query structure with one or several operators which contain axis generalization, leaf deletion and subtree promotion described in the next section. Then we expand subject among all the domain ontology descend by the correlations. Second, expand the predicate or the object based on the subject and compute the similarity. Third, we reason the rest component and compute the similarity. Final we compute the similarity of the expansion queries and select the ones above the threshold.

3.2.1 Structure Expansion

Given a user query, we would like to generate relaxed queries, evaluate them and return a ranked list of answers. A significant challenge in this is how to relax query structure. We need a systematic way to generate queries that are guaranteed to be relaxations and cover all relaxations of structure.

Now we present a set of operators for this purpose. The three operators are specific to query structure.

Axis Generalization: An ancestor-descendant (represented as “ad”) relationship can generalize from a parent-child (represented as “pc”), and they associated in some way. More precisely, if there is a tree query Q and a predicate $pc(\$x; \$y)$ in the logical representation of query. $O1_{pc(\$x, \$y)}(Q)$, the axis generalization of Q on $pc(\$x; \$y)$, is the tree query which is identical to Q except the pc-edge from $\$x$ to $\$y$ in T is replaced with an ad-edge from $\$x$ to $\$y$.

Leaf Deletion: The intuition behind this operator is that if we delete a leaf node $\$x$ in the query, we allow answers where that leaf node might not be matched. More precisely, given a query tree Q and a leaf node $\$x$ in query tree, $O2_{\$x}(Q)$, the leaf deletion of Q on $\$x$, is the query tree which is identical to Q except the leaf node $\$x$ is deleted from original tree. In order to avoid queries that evaluate to true on every element, we forbid deleting the root of a TPQ (Tree Pattern Queries). Taking $Q1$ as an example, $O2_{\$x}(Q1)$ applied to $Q1$ are $Q2$ and $Q4$. Query $Q5$ is an extreme case where leaf node deletion is applied repeatedly on the user query.

Subtree Promotion: This permits a query subtree to be promoted so that the subtree is directly connected to its former grandparent by an ad-edge. More precisely, let Q be a query tree, $\$x$ any node of Q other than the root, and $\$y$ its grandparent in Q . Then $O3_{\$x}(Q)$, the subtree promotion of Q on $\$x$, is the new query tree identical to Q except the subtree rooted at $\$x$ is made a corresponding subtree of $\$y$, where the edge between $\$y$ and $\$x$ is an ad-edge. As an example, $O3_{\$3}(Q1)$ applied to query $Q1$ results in the query $Q3$ in Figure 3.

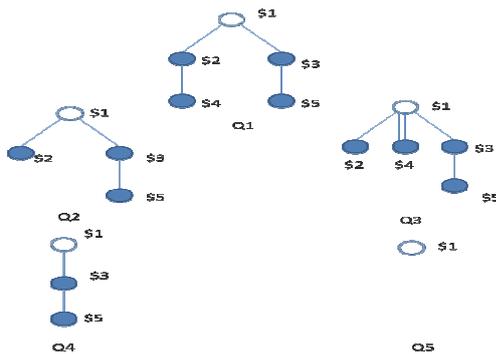


Figure 3: Query tree.

Every query obtained by applying a composition of one or more of the operators $O1$, $O2$, $O3$ applied to Q is a valid structural relaxation. Every valid relaxation of Q can be obtained by many applications of these operators to Q . To ensure the

completeness of query structure relaxation, we have a mechanism to systematically generate all and only valid relaxations of a given tree pattern query.

3.2.2 Vocabulary Expansion

Vocabulary expansion contains not only synonyms, but also hypernyms and hyponyms. WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. WordNet provides relations beyond is-a, including has-part, is-made-of, and is-an-attribute-of. WordNet interlinks not just word forms—strings of letters—but specific senses of words. As a result, words that are found in close proximity to one another in the network are semantically disambiguated. WordNet labels the semantic relations among words, whereas the groupings of words in a thesaurus do not follow any explicit pattern other than meaning similarity.

WordNet::Similarity is a freely available software package that makes it possible to measure the semantic similarity and relatedness between a pair of concepts (or synsets). WordNet::Similarity implements measures of similarity and relatedness that are all in some way based on the structure and content of WordNet. We expand the vocabulary of query terms with the similarity score above a certain threshold.

3.3 SPARQL Query Construction

After query expansion, we need transfer them into queries to retrieve information. SPARQL is a query language and data access protocol for RDF data model, and now has become a W3C Recommendation. SPARQL query language achieves the query function through a graphical mode (Graph Pattern) matching. The simplest graphical mode is the mode of the triples and triple mode allows the query variable appearing in the subject, predicate, or object position.

4 SEMANTIC SIMILARITY COMPUTATION

4.1 Hierarchical Structure Similarity

As the relaxation operators defined in section 3.2.1, all other relaxation structure can be achieved through composition of one or more of the operators.

Each operation results in similarity reduction between the changed and the original RDF triples in structure aspect. We assume that each operator of O1, O2, and O3 results in reduction of the same similarity. We exploit a mechanism to systematically generate all and but valid relaxations of a given tree pattern query. We list all the structure relaxations and the number of the operations. Then we use following formula to normalize the relaxation queries similarity to [0, 1].

$$HSsimi(q_i) = 1 - \frac{n_i}{\max(n_i) + 1} \quad (1)$$

n_i represents the number of operations through which expansive query i vary from original query.

4.2 Vocabulary Similarity

WordNet::Similarity provides six measures of similarity, and three measures of relatedness, all of which are based on the lexical database WordNet (<http://wordnet.princeton.edu>). These measures take two concepts as input, and return a numeric value that represents the degree that they are similar or related to each other.

Five different proposed measures of similarity or semantic distance in WordNet are experimentally compared by examining their performance in a real-world spelling correction system. It was found that Jiang and Conrath's measure (Jiang, 1997) gave the best results overall (Budanitsky, 2001). So we choose JCN as semantic similarity measures.

As an element in <subject, predicate, object> is not necessary word, it is more likely to be a phrase. The name similarity between the two sets of name phrase T1 and T2 can be determined as the average best similarity of each token with all tokens in the other set. So we use formula (2) to compute the phrase similarity as follows:

$$P_{sim}(T1, T2) = \frac{\sum_{t1 \in T1} [\max_{t2 \in T2} sim(t1, t2)] + \sum_{t2 \in T2} [\max_{t1 \in T1} sim(t2, t1)]}{|T1| + |T2|} \quad (2)$$

$t1$ represent the word in T1, $t2$ represent word in T2.

$P_{sim}(t1, t2)$ measure the relatedness of $t1$ and $t2$ using JCN. To compute the credit of expensive RDF triples, we adapt the formula below. Because of the probability of predicate or object absence, we adapt both multiply and add in the formula (3). CreditS, CreditP and CreditO are respectively the similarities between the expansive subject, predicate, object and the input ones. CreditTriples is the similarity between expensive RDF triples and input RDF triple.

$$CreditTriples(q_i) = \alpha CreditS * CreditP * CreditO + \beta CreditS + \gamma CreditP + \theta CreditO \quad (3)$$

$\alpha, \beta, \gamma, \theta$ represent the weight of each component.

And $\alpha + \beta + \gamma + \theta = 1$.

We finally combine the structure similarity with vocabulary similarity to credit the final relaxation query scores.

5 EVALUATION

In this section, we will introduce our system with an example scenario, and then we compare our system with classical text retrieval system. Our query system has an appropriate usage scenario—IOT. The Schema of IOT XML is of severe heterogeneity. To build the IOT ontology, we collect xml documents from six websites of IOT platforms, which contain Pachube (<https://xively.com/>), Cloudsensing (<http://wot.cloudsensing.cn>), Evrythng (<http://www.evrythng.com/>), Thingspeak (<http://thingspeak.com/>), Yeelink (<http://www.yeelink.net/>) and Exosite (<http://exosite.com/>). The xml documents contain 24,782 xml files. We run the experiment on an Intel Core 2 Duo 2.8GHz ma-chine with 1536MB memory and 250GB disk space.

5.1 Example Scenario

As an example, the user wants to know the maker of temperature sensor located in Beijing. So he writes “Who is the maker of temperature sensor which locates in beijing?” in the searchBox. The figure of the structure above is figure 2.

Depended on the grammar dependency tree, we can abstract the RDFs (temperature sensor, maker, who) and (temperature sensor, lactation, beijing) combining with our triple RDF abstraction rules and templates. The query tree shows as follow:

Then we relax the query tree structure as mentioned in the section of *Hierarchical Structure similarity*. Then we expand our subject, predicate and object referring to the section of *Vocabulary Similarity*. We compute the similarity of the expansive triples and original RDF triple.

If you are a specialist of IOT, you may know that the temperature and humidity are generally integrated in one device. And sometime they publish only one manufacturer. This can be reasoned from the domain ontology. We should modify the triple to (temperature/humidity sensor, manufacturer, person) based on the domain ontology. We can make similar

Search Result

Abstract triple: (temperature sensor, maker, who) (temperature sensor, location, beijing)
the search costs 9031 ms

| Subject | Predicate | Object | Predicate | Object | Related node |
|--------------------|--------------|-------------------|-----------|----------|-------------------------------------|
| temperature sensor | manufacturer | Beijing konglun | location | Beijing | more relative nodes |
| temperature sensor | manufacturer | GRAINGER | location | Shenzhen | more relative nodes |
| temperature sensor | manufacturer | shuanghai jiekong | location | Shanghai | more relative nodes |
| humidity sensor | manufacturer | GRAINGER | location | Beijing | more relative nodes |
| wind power sensor | manufacturer | Beijing konglun | location | Beijing | more relative nodes |

Figure 5: The first 5 results.

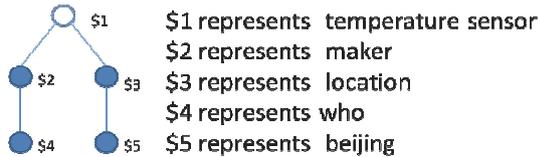


Figure 4: Query tree.

reasoning by adding expert knowledge to the domain ontology. Then translate these triples to SPARQL sentences to retrieve the RDF data. The first 5 results is as the Figure 5, Click any the “more relative nodes”, we can redirect to see the parent, siblings and children nodes.

5.2 The Evaluation of Our System

- The precision of our system

Then we construct 363 queries by 34 testers in the forms of both natural language and keyword. We do information retrieval in both keyword search system and our system. The keyword search was implemented like the text retrieval system. The precise evaluation is as in the figure 6.

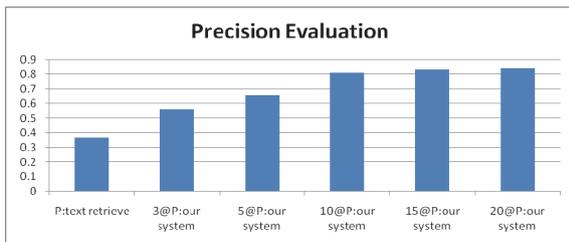


Figure 6: The precision.

We can get a very high precision in first 10 results. Because of the insufficiency of WordNet itself, we lose about 10% precision.

- Query Execution Time

In order to determine the scalability of our system, we check how long it on average takes to accomplish a search. Because the scale of domain ontology is usually fixed without much fluctuation, so the time will not increase linearly with the xml

document. Now we conduct an experiment to see the relation of similarity threshold and searching cost time.

Almost all the searching cost is no more than one second. The search time is mainly decided by the scale of the ontology and the threshold of similarity expansion. As figure shows, as the threshold reduces, the searching cost time declines.

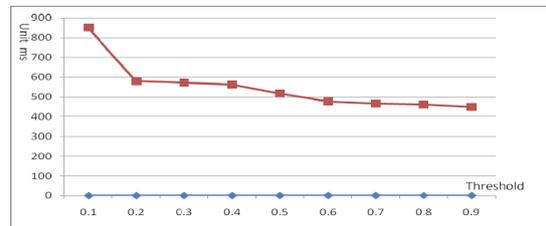


Figure 7: Query execution time.

6 CONCLUSIONS

This paper proposes an xml semantic query system that combines general ontology like WordNet with certain domain ontology constructed by RDF Schema. We consider two aspects of query relaxation, vocabulary and hierarchical structure, to ensure the relevance sort and the recall of results. We rely on natural language processing techniques to provide user-friendliness, allowing NL input query. In this paper, I have summarized some of the structure patterns to extract RDF queries from most common query statements. This approach has limitations of corpus scale and accuracy of RDF extraction. In future work, we can use machine learning, and even deep learning to enhance the accuracy of extracting RDF.

ACKNOWLEDGEMENTS

The State Key Program of China- project on the Architecture, Key technology research and

Demonstration of Web-based wireless ubiquitous business environment (2012ZX03005008).

REFERENCES

- Guerrini G. 2013. Approximate XML Query Processing[M]//Advanced Query Processing. Springer Berlin Heidelberg.
- Y. Li, C. Yu, H.V. Jagadish. 2004. Schema-free xquery, in: *Proceedings of the 30th International Conference on Very Large Data Bases (VLDB2004)*.
- S. Cohen, J. Mamou, Y. Kanza, Y. Sagiv. 2003. Xsearch: a semantic search engine for XML. in: *Proceedings of 29th International Conference on Very Large Data Bases (VLDB2003)*.
- G. Li, J. Feng, J. Wang, L. Zhou. 2007. Effective keyword search for valuable lcas over XML documents. in: *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management*.
- Liu X, Chen L, Wan C, et al. 2013. Exploiting structures in keyword queries for effective XML search[J]. Information Sciences.
- Kaufmann E, Bernstein A. 2010. Evaluating the usability of natural language query languages and interfaces to Semantic Web knowledge bases[J]. Web Semantics: Science, Services and Agents on the World Wide Web.
- Decker S, Melnik S, Van Harmelen F, et al.2000. The semantic web: The roles of XML and RDF[J]. Internet Computing.
- J. Xu, W. Li. 2007. Using Relational Database to Build OWL Ontology from XML Data Sources. *Proceeding 2007 International Conference on Computational Intelligence and Security Workshops, CISW*.
- Liu. Z, Chen. Y. 2008. Reasoning and identifying relevant matches for XML keyword search. PVLDB.
- Y. Xu, Y. Papakonstantinou. 2005. Efficient keyword search for smallest lcas in XML databases. in: *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD2005)*.
- Golenberg K, Kimelfeld B, Sagiv Y. 2008. Keyword proximity search in complex data graphs[C]. *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*.
- Bao Z, Lu J, Ling T W.2010. Xreal: an interactive xml keyword searching[C]//*Proceedings of the 19th ACM international conference on Information and knowledge management. ACM*.
- Theobald M, Bast H, Majumdar D, et al. 2008. TopX: efficient and versatile top-k query processing for semistructured data[J]. *The VLDB Journal*.
- Amer-Yahia S, Koudas N, Marian A, et al. 2005. Structure and Content Scoring for XML. In: VLDB.
- Augsten N, Barbosa D, Bohlen M, et al. 2010. Tasm: Top-k approximate subtree matching[C]//Data Engineering (ICDE), 2010 *IEEE 26th International Conference on*.
- Sanz I, Mesiti M, Guerrini G, et al. 2008. Fragment-based approximate retrieval in highly heterogeneous XML collections[J]. Data & Knowledge Engineering.
- De Marneffe M C, Manning C D. 2008. Stanford typed dependencies manual[J].
- J. Jiang and D. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*.
- Budanitsky, A., & Hirst, G. 2001. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Workshop on WordNet and Other Lexical Resources*.

APPENDIX

Triples extraction principle is based on the whole dependencies between words , vocabulary speech , marked entity to mark RDF triples .

Module 1 : Get subject dependencies, analyze the subject

- 1)Dependencies extract is satisfied {"nsubj", "xsubj", "top", "nsubjpass"} in any one of dependencies.
- 2)Get the current dependency of entities as the center of the word , when analyzing the association triples.
- 3)If the subject followed with Prep_of structure, the nouns after the Prep_of structure become the subject, and the nouns before degenerate to predicate.
- 4)Transfer the dependency nodes into triple nodes

Module 2 : Get relations of predicate and object, analyze predicate and object

- 1)Extract dependencies that satisfy {"dobj", "pobj", "range", "attr", "dep", "ccomp", "pccomp", "lccomp", "rcomp"} or {"conj", "ccomp", "pcomp", "lcomp", "rcomp"} dependencies. If the predicate is nonsense word like is, ignore it.
- 2)According to the word collection of entity types types, access to the current node as the center of the " predicate - object" right, first obtain the predicate , and then get the object.
- 3)Adjustment predicate, object because the original object may be predicate of clause and the subject of clause may be predicate of the main clause)
 - Analysis predicate : take the current word as predicate in the center to analyse the association triples
 - Analysis of object : Get the current dependency of entities as a subject and the center of the word , when analyzing the association.
- 4) If the predicate is a verb, translate it to noun.

Module 3: Connect predicate and object with subject to RDF triples

Finding Domain Experts in Microblogs

Shao Xianlei, Zhang Chunhong and Ji Yang

*Mobile Life and New Media Laboratory, Beijing University of Posts and Telecommunications (BUPT), Beijing, China
shaoxianlei@163.com, {zhangch.bupt.001, ji.yang.0001}@gmail.com*

Keywords: Domain Experts Finding System, Microblog Lda, GBDT, User Features.

Abstract: As users and contents of microblogging services gain a sharp increase, it presents the challenge of finding domain experts who are of high profession but generally don't have followers widely. To address this, we propose a domain experts finding system, which consists of three modules: data preprocessing module, user features extracting engine, experts identifying and ranking module. Firstly, we extract three kinds of features for characterizing social media authors, including user profile features, tweeting behavior features and linguistic content features which are generated by our Microblog Latent Dirichlet Allocation(Microblog Lda) model. Secondly, by casting the problem of finding domain experts as a 0-1 classification problem, we use the Gradient Boosted Decision Trees (GBDT) framework to do probabilistic classification over these features, execute a ranking procedure and yield a list of top N users for a given domain. Experimental results on actual datasets show our Microblog Lda outperforms LDA(Latent Dirichlet Allocation) and our system has a high accuracy in the task of finding domain experts in Microblogs.

1 INTRODUCTION

Millions of people turn to microblogging services such as twitter which is known to all and Sina Microblog which is the most influential microblogging services in China to gather real time news or opinions about people, things, or events of interest. Such services are not only used as social networking to stay in touch with friends and colleagues but also used as publishing platforms to create and consume content from sets of users with overlapping or disparate interests.

Through a survey on users' following decisions on Twitter (Ramage, 2010), we can know that the most two common reasons for users to make following decisions are "professional interest" and "technology". From this conclusion and our long-term observation of user behavior, it is not difficult to find that meeting users' demand to access domain expertise of users would make a great significance for both the advancing of microblogging services and the efficiency of using microblogging.

In order to meet users' demand to access expertise, finding the users that are recognized as sources of relevant and trustworthy information in specific domains is an important challenge. But currently, Twitter and Sina Microblog interface fails to support such kinds of services.

Despite the important role of domain expert users in microblogging, the challenge of identifying true experts is trickier than it appears at first blush. Content in microblogging systems is produced by tens to hundreds of millions of users. In microblogging contexts, for any given domain, the number of these content producers even in a single day can easily reach tens of thousands. While this large number can generate notable diversity, it also makes finding the true experts, those generally rated as learned and authoritative in a given domain, challenging.

Furthermore, most domain experts are not as well known as some celebrities known by many people, they are less discoverable due to low network metrics like follower count and the amount of content produced to date. Thus, we cannot use traditional graph-based methods of discrimination degree of user authority to find domain experts. Besides, graph based algorithms are computationally infeasible for near real time scenarios (Pal, 2011) and social graph information has a negligible impact on the overall performance of identifying a user (Pennacchiotti, 2011).

In this paper, we propose a new method for finding domain experts in microblogs. To sum up, the contributions of this paper are: (1) we propose a domain experts finding system which can identify

true experts in Microblogs with high accuracy. (2) A user feature engine is build to extract user features that are useful to identify one's authority. (3) Microblog Lda, which is based on Lda (Blei, 2003) but is more suitable for microblogging-style informal written genres, is proposed to extract users' linguistic content features.

The rest of the paper is organized as follows: Section 2 places our research in the context of previous work. Section 3 gives the framework of our domain experts finding system. Details of each module of our system are provided separately in Section 4 and Section 5. Results of experiments, which are provided in Section 6, show that the Microblog Lda can obtain significant performance gains and the system, as a whole, can achieve high accuracy in finding true experts in a given domain.

2 RELATED WORK

Within the microblogging research field, little work has explored the issue of domain expert identification. There have been several attempts to measure the influence of Twitter users and thereby identify influential users or experts (Bakshy, 2011; Cha, 2010; Romero, 2011). To our knowledge, there have been only two notable efforts that have approached the problem of identifying experts in specific topics (Weng, 2010; Pal, 2011). (Weng, 2010) proposed a Page-Rank like algorithm TwitterRank that uses both the Twitter graph and processed information from tweets to identify experts in particular topics. On the other hand, (Pal, 2011) used clustering and ranking on more than 15 features extracted from the Twitter graph and the tweets posted by users.

While somewhat similar to paper (Pal, 2011), our method differs in several important ways. Firstly, in paper (Pal, 2011), authors only emphasized users' tweeting behavior features but ignored the precise linguistic content features which can make great significant to domain experts finding task. In our paper, we choose several features used in (Pal, 2011) which are suitable for our target users – Sina Microblog users but also add some more features. Secondly, apart from users' tweeting behavior, we also make use of users' profile features and linguistic content features and use a new method to build the features of users. Finally, our approach offers the potential advantage over network-based calculations in that it is less likely to interface by a few users with high popularity (i.e., celebrities).

Outside microblogging, finding authoritative users generally has been widely studied. Authority finding has been explored extensively on the World Wide Web. Amongst the most popular graph based algorithms towards this goal are PageRank, HITS and their variations (Page, 1998; Kleinberg, 1998; Farahat, 2002). Also predating microblogging, several efforts have attempted to surface authoritative bloggers. (Java, 2006) model the spread of influence on the Blogosphere in order to select an influential set of bloggers which maximize the spread of information on the blogosphere.

Authority finding has also been explored extensively in the domain of Community question answering (CQA). Among most of the models proposed, some authors used network modeling approach (i.e., Agichtein, 2008). Others modeled CQA as a graph induced as a result of a users' interactions with other community members (Jurczyk, 2007; Zhang, 2007). Still other approaches used characteristics of users' interactions (Bougoussa, 2008, Pal, 2010).

In the domain of academic search, authority identification also has been studied extensively. (Tang, 2008) studied the problem of expertise search in their academic search system-ArnetMiner. (Kempe, 2003) modeled the spread of influence in co-authorship networks.

Summarizing related work, the problem of finding authority has been explored extensively in other domains. Among these work, some used network analysis approaches which is computationally expensive, some used structured information (i.e., users' interaction behaviors) and some used both approaches in an integrated way. Our domain of interest, microblogging, has seen far less attention. As mentioned above, we feel our approach extends research in the following points: apart from users' interaction behaviors, we also use users' linguistic content features which carry rich information about users; without using graph-based approach, we use a classification approach which is computationally tractable.

3 DOMAIN EXPERTS FINDING SYSTEM

Our domain experts finding system mainly consists of three parts: data preprocessing module, user features extracting engine, experts identifying and ranking module. The framework of our system is shown in the following Figure 1.

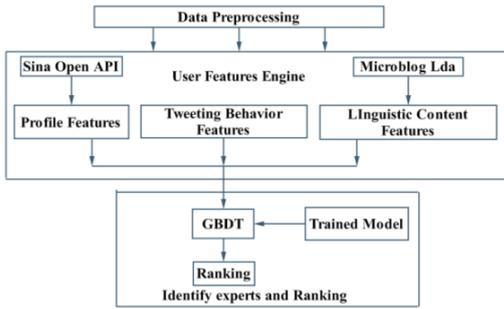


Figure 1: Framework of Domain Experts Finding System.

The work of data preprocessing module is to prepare cleaned source data for features engine and experts identifying and ranking module. Details of this module's workflow are described in Section 6.

In our proposed system, user features extracting engine can automatically construct user features and extract numerous features that are useful in domain expert authentication. In Section 4, we will describe the details of user features extracting engine and give a comprehensive analysis to the features we choose.

In Section 5, we would describe how we use the features extracted in Section 4 in our classification model to identify experts. The module will eventually generate the experts list and give the top N experts.

4 USER FEATURES EXTRACTING ENGINE

To learn the classification model, we use a large set of features that can reflect the impact of users in the system and their expertise. According to the nature they aim to capture, the features can fall into three main categories: profile features, tweeting behavior features and linguistic content features.

The rest of this section will further describe in depth these main categories of user features.

4.1 Profile Features

To start we present the list of valuable profile features in Table 1.

Having registered the service, users would have several profile features such as PF1-5 which are maintained by the microblogging service system automatically. Through the open API (application program interface) service of microblogging, we can get these profile features of users.

Experimental, a domain expert is more likely to

Table 1: Profile Features.

| Name | Feature |
|------|----------------------|
| PF1 | Followers Count |
| PF2 | Verified |
| PF3 | Friends Count |
| PF4 | Statuses Count |
| PF5 | Favorites Count |
| PF6 | Followers per Friend |
| PF7 | Description Score |
| PF8 | Tags Score |

have higher PF1, PF4 and PF6 because of his identity of information provider. PF2 is a service provided by microblogging system. If a user is authenticated, his identity is more likely to be true.

In self-descriptions and tags, users would like to use some words or sentences to describe themselves and choose tags provided by microblogging system to stand for them. Hence, from users' descriptions and tags we can partially know their interests and domains. In this paper, we convert user's description and tags to two features, PF7 and PF8. By counting words used in description of training users in the domain we care, we get top N words in all users' descriptions according to their word frequency, which is expressed as D_{domain} . PF7 is calculated using formula (1).

$$PF7 = \frac{|D_i \cap D_{domain}|}{|D_{domain}|} \quad (1)$$

Where D_i is the words in i th user's descriptions.

Similarly, PF8 is calculated using the following formula (2).

$$PF8 = \frac{|T_i \cap T_{domain}|}{|T_{domain}|} \quad (2)$$

Where T_{domain} is top N tags in all users' tags with high frequency and T_i is tags of the i th user.

4.2 Tweeting Behavior Features

Tweeting behavior is characterized by a set of statistics capturing the way the user interacts with the microblogging service. In paper (Pal, 2011), the authors listed several tweeting behavior features that reflect the impact of users in microblogging system. In our paper, we use some of features that listed in paper (Pal, 2011), and add more features that can be extracted from Sina Microblogging service. The valuable tweeting behavior features we used are listed in Table2.

In paper (Java, 2007), the authors suggested

that users who often post URLs in their tweets are most likely information providers. Giving an URL in microblogs is an efficient way to supply information in depth. In our work, we use feature TBF1 to record number of links user shared.

Hashtag keywords (TBF2) are words starting with the # symbol and are often used to denote topical keywords in microblogs. These keywords can clearly reflect the topic of microblog.

Table 2: Tweeting Behavior Features.

| Name | Feature |
|-------|---|
| TBF1 | Number of links shared |
| TBF2 | Number of keyword hashtags(#) used |
| TBF3 | Number of conversation microblogs |
| TBF4 | Number of retweeted microblogs |
| TBF5 | Number of mentions (@) of other users by author |
| TBF6 | Number of unique users mentioned by the author |
| TBF7 | Number of users mention by the author |
| TBF8 | Average number of messages per day |
| TBF9 | Average comments per microblog |
| TBF10 | Average reports per microblog |

In paper (Boyd, 2010), retweeting or reposting someone's post were discussed. A user can mention other users using the "@user" tag. In paper (Honeycutt, 2009), authors discussed @user. And in papers (Naaman, 2010) and (Ritter, 2010), authors modeled the conversations. It's not difficult to know that features TBF2-7 can make a big difference in identifying domain experts. As an information provider, a domain expert tends to tweet several or even dozens of messages a day. TBF 8 can measure the impact of this behavior. Because the content of microblogs tweeted by domain experts is of high value, follows of experts would comment or even repost it. Statistics show that the higher the features TBF9 and TBF10 are, the higher user's authority is.

4.3 Linguistic Content Features

According the results in paper (Pennacchiotti, 2011), user's microblogs content makes most of the contribution in user features extraction. Making a good use of microblogs content would determine the performance of our system in a large extent.

Linguistic content information encapsulates the user's behavior of lexical usage and the main topics the user is interested in. Several studies, e.g. (Rao, 2010), have shown that bag-of-words models usually outperform more advanced linguistic ones.

Different from other primarily spoken genres previously studied in the user-property classification literature, microblogging-style informal written genres has its own characteristic.

The content of microblog can fall into three categories: original microblog, which is produced by the author; conversation microblog, which is replied by the author; reposted microblog, which is produced by someone else and forwarded by the author with some additional comments. In Sina Microblog service, the format of conversation microblog and reposted microblog is shown as follow:

Conversation microblog:

回复 (reply)@user: content of reply//@user: source content.

Reposted microblog:

Additional comments //@user: source content.

4.3.1 Microblog Latent Dirichlet Allocation

Reply and repost characterize the relation between microblogs. In general, content of reply in conversation microblog and additional comments in reposted microblogs shares related topics with source content of microblog. In this paper, we take into account the above two relationships, extend the original Lda (Blei, 2003), and propose our Microblog Lda.

Microblog Lda adopts the basic idea of topic model, namely each microblogging exhibits multiple topics which are represented by probability distributions over words, denoted as $P(z|w)$ respectively. The Bayesian network of Microblog Lda is shown as follow in Figure 2.

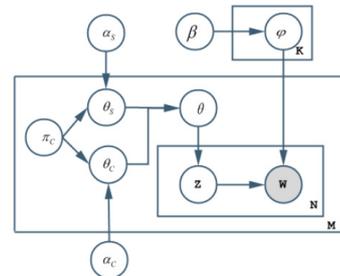


Figure 2: Bayesian network of Microblog Lda.

Apart from special instructions, symbols in Microblog Lda follow the definitions in (Blei, 2003).

Microblog Lda generates microblogging in the following process:

- 1 . Random choose a topic distribution over words.
- 2 . Judge whether a microblogging is retweeted or replied. If so, mark π_c as 1, random choose a contactor-topic distribution θ_c , which is sampled from a Dirichlet distribution with

hyperparameter α_c , then assign the value of θ_c to θ_s ; if not, random choose a document-topic distribution θ_s , whose id sampled from a Dirichlet distribution with hyperparameter α_s . The probability distribution of θ is shown as follows:

$$\begin{aligned} P(\theta; \alpha) \\ &= P(\theta; \alpha, c) \\ &= P(\theta_c; \alpha_c)^{\pi_c} P(\theta_s; \alpha_s)^{1-\pi_c} \end{aligned} \quad (3)$$

3. Draw the specific word w_{dn} from the Multinomial distribution with parameter $\varphi_{z_{dn}}$.

For a microblogging, the joint probability is :

$$\begin{aligned} P(W, Z, \theta, \varphi, \alpha, \beta) = \\ \prod_{i=1}^K P(\varphi_i; \beta) \times \\ \prod_{j=1}^M P(\theta_j; \alpha_c)^{\pi_c} P(\theta_j; \alpha_s)^{1-\pi_c} \times \\ \prod_{t=1}^N P(W_{j,t} | \varphi_{z_{j,t}}) P(Z_{j,t} | \theta_j) \end{aligned} \quad (4)$$

Generative process is shown as follows:

Algorithm 1: Microblog Lda.

```

For each topic  $k \in \{1, 2, \dots, T\}$  do
  Draw  $\varphi_k \sim Dir(\beta)$ 
End for
For each microblog d do
  Judge whether d is conversation or reposted
  microblog
  If true
    Draw  $\theta_s = \theta_c \sim Dir(\alpha_c)$ 
  Else
    Draw  $\theta_s \sim Dir(\alpha_s)$ 
  For each word  $w_{dn}$  do
    Draw  $z_{dn} \sim Multi(\theta_s)$ 
  End for
End for

```

4.3.2 Topic Features

Our Microblog Lda model is an adaptation of the original Lda proposed in paper (Blei, 2003), where documents are replaced by user's stream. Our hypothesis is that a user can be represented as a multinomial distribution over topics. While (Blei, 2003) represents documents by their corresponding bag of words, we represent users in microblogging

service by the words of their tweets.

Results from (Pennacchiotti, 2011) shown that Lda system outperforms the tf-idf baseline with statistical significance. These prove our claim that topic models are good representations of user-level interests.

User's multinomial distribution over topics can clearly reflect his interest. Therefore domain experts' multinomial distribution over topics would be distinct. In our paper, we used results of Microblog LDA as linguistic content features of user and modeled each user by a topic-vector, where the weights are the probabilities to emit the topic.

5 EXPERTS IDENTIFYING AND RANKING

In Section 4, we generated user features, including profile features, tweeting behavior features and linguistic features, using our user features engine. In this section, we would use features generated above to identify domain experts and rank the result list.

In this paper, we cast the problem of identifying domain expert as a problem of 0-1 classification. As a classification algorithm, we use the Gradient Boosted Decision Trees – GBDT framework (Friedman, 2001). (Friedman, 2001) shows that by drastically easing the problem of over-fitting on training data (which is common in boosting algorithms). GBDT outperforms the state-of-the-art machine learning algorithms such as SVM with much smaller resulting models and faster decoding time (Friedman, 2006).

We use the features listed in section 4 to learn the classification model. After learning the GBDT model, we will use it to classify the large set of Sina Microblogging users and give the probability of a user judged as a domain expert.

In GBDT framework, results are shown in the format of probability of a user classified into classes. Having generated the probability of a user seen as a domain expert, we can ranking the probability and give the top N most liked experts of the domain we care.

6 EXPERIMENTAL EVALUATION

6.1 Data Preprocessing

Different from English, there are no spaces in words

interval of Chinese sentences. In order to process Chinese data, we should firstly segment sentences into words. In this paper, we use the ICTCLAS Chinese word segmentation system which has a high accuracy in Chinese word segmentation.

After word segmentation, we would discard all words that appear in a stop-word dictionary.

During July 1-15, we invited a pool of experts and seniors in the field of open source hardware. Through collecting their opinions extensively, we choose 200 users to train and validate our domain experts finding system, among them 92 are experts in open source hardware domain and 108 are not experts in open source hardware domain.

To train Microblog Lda model, we crawled all microblogs of these 200 users on Sina Microblog which is a microblogging service in China like twitter. There are 428 thousand microblogs totally.

6.2 Effectiveness Experiment

6.2.1 Performance of Microblog Lda

We conducted the comparative experiment between Microblog-lda and Lda using perplexity, measure of performance for statistical models which indicates the uncertainty in predicting a single word.

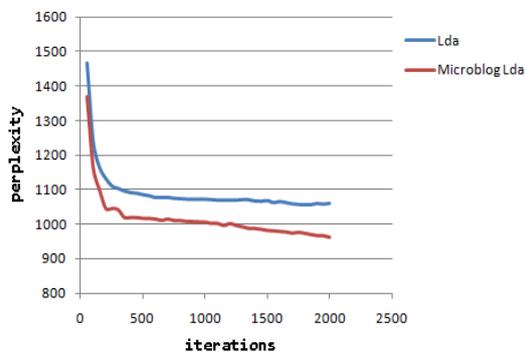


Figure 3: Perplexity of Lda and Microblog Lda.

Perplexity is used to measure the performance of LDA and Microblog-lda under the same hyperparameters setup, and the result is shown in Figure 3. From the result in Figure 3, we can see that Microblog Lda has plenty of performance gains compared with Lda.

6.2.2 Performance of Domain Experts Finding System

We compared our model with two baseline models as described below.

Baseline1: In this model, we used features listed in (Pal, 2011) only. Then, these features is used in our domain experts finding system and to give results on our dart base.

Baseline2: In this model, we used users' linguistic content features only.

Our: we used all kinds of features as mentioned above, including profile features, tweeting behavior features, linguistic content features.

After data processing and feature extraction, classification approaches are employed based on GBDT framework. The result is obtained with 10-fold cross validation in Figure 4. In this paper, we use ROC Area which refers to the area under ROC curve to measure the quality of our classifier and F-measure to measure the accuracy of our classifier comprehensively. We also give the results of Precision and Recall.

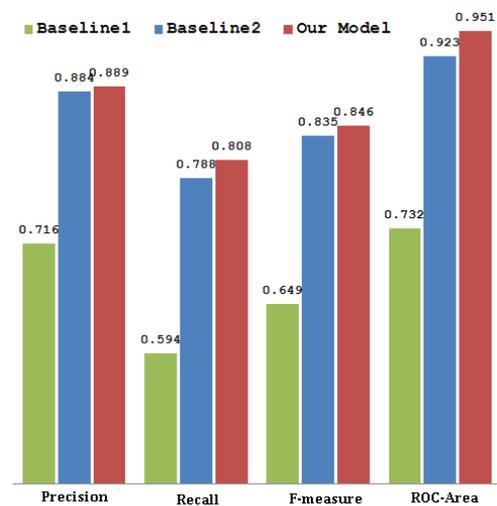


Figure 4: Classification results of training dataset.

In the results of our experiments, we give the performance comparisons of our domain experts finding system with baseline1 and baseline2. Compared with baseline1, both baseline2 and our model gain a great increase in performance. In Figure 4, we can know that linguistic content features are highly valuable and contribute most of the classification confidence. From the index of ROC Area, we can know that our domain experts finding system is of high quality. From the index of Precision and F-measure, we can know that our domain experts finding system has the ability to find experts in a particular domain with high accuracy.

6.2.3 Experts Identifying and Ranking

In order to test performance of our system in real

production environment, we searched microblogs using keywords –“open source hardware” in search engine of Sina Microblog. The search engine would return the microblogs which content our search keywords. All microblogs were published recently. After parsing the returned microblogs and extracting the user id in the microblogs, we obtained initial users list which contents users who are likely to be expert in open source hardware domain. In our experiments, there were 3934 users in the users list.

Next, we used our domain experts finding system to analysis these users and identified 46 users who can be recognized as experts. In table 3, we give top 10 users in the domain of open source hardware. In order to compare preformance of our domain experts finding system with existing system, in table 4 we give top10 users returned by People Search System of Sina Microblog using keyword “open source hardware”.

Table 3: Top 10 users returned by domain experts finding system.

| Id | Screen Name |
|------------|--------------------------------------|
| 2171581500 | SeedStudio |
| 2305930102 | 柴火创客空间(Chai huo chuang ke kong jian) |
| 2524468112 | Arduinos |
| 3160959662 | KnewOne |
| 2055985387 | 王盛林 Justin(Wang sheng lin Justin) |
| 3657027664 | 开放制造空间(Kai fang zhi zao kong jian) |
| 1683765255 | 导通不能(Dao tong bu neng) |
| 1906419177 | 新车间(Xin che jian) |
| 1497878075 | 老黄(Lao huang) |
| 1518434112 | 李大维(Li da wei) |

In top 10 users returned by People Search System of Sina Microblog, the former six users’s name have the search keyword “open source hardware”. This means that People Search System of Sina System currently can not search out experts accurately, such as, a common user has screen_name containing the keywords, his is more likely to be returned.

In the users returned by our domain experts finding system, their have real people and organization family. Specially, in order to evaluate the performance of our system, we made a questionnaire survey on 20 members of a club which focuses on open source hardware. From the feedback of these interviewees, we can get that 91.5% of users returned by our domain experts finding system can be recognized as experts in the particular domain.

Table 4: Top 10 users returned by People Search System of Sina Microblog.

| Id | Screen Name |
|------------|--|
| 1750097377 | 开源硬件的星星之火(Kai yuan ying jian de xing xing zhi huo) |
| 2334652932 | 赛灵思开源硬件社区(Sai ling si kai yuan ying jian she qu) |
| 2497494380 | 开源硬件(Kai yuan ying jian) |
| 3561629704 | 小米开源硬件俱乐部(Xiao mi kai yuan ying jian ju le bu) |
| 2356441795 | 开源硬件平台 |
| 1906419177 | 新车间(Xin che jian) |
| 2284986847 | 北京创客空间(Bei jing chuang ke kong jian) |
| 2305930102 | 柴火创客空间(Chai huo chuang ke kong jian) |
| 1715452481 | 54chen |
| 1518434112 | 李大维(Li da wei) |

7 CONCLUSIONS

In this paper, we proposed a domain expert finding system that could be used to produce a list of top N domain experts in Microblogs. We showed that: the thought of casting the problem of finding domain experts to a problem of 0-1 classification is feasible and of high accuracy in practice. From our experimental results, we can know that our domain experts finding system achieves good performance. In this paper, we use three kinds of user features, including profile features, tweeting behavior features and linguistic content features. Among them, linguistic content features show especially robust performance across tasks.

For further work, we wish to explore in detail running our system in parallel computing platform, like Hadoop. In addition, we wish to explore in detail how different features affect the final ranking and eliminate the influence of negative features.

ACKNOWLEDGEMENTS

The State Key Program of China- project on the Architecture, Key technology research and Demonstration of Web-based wireless ubiquitous business environment (2012ZX03005008).

REFERENCES

- A. Java, P. Kolari, T. Finin, and T. Oates. 2006. *Modeling the spread of influence on the blogosphere*. In WWW (Special interest tracks and posters).
- A. Java, X. Song, T. Finin and B. Tseng. 2007. Why we

- twitter: understanding microblogging usage and communities. *Joint 9th WEBKDD and 1st SNA-KDD Workshop (WebKDD/SNA-KDD)*.
- A. Pal and J. A. Konstan. 2010. Expert Identification in Community Question Answering: *Exploring Question Selection Bias*. In CIKM.
- A. Ritter, C. Cherry and B. Dolan. 2010. Unsupervised Modeling of Twitter Conversations. *In the 2010 Conference of the North American Chapter of the Association for Computational Linguistics*.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. 2003. Latent dirichlet allocation. *the Journal of machine Learning research*.
- C. Honeycutt, S. C. Herring. 2009. Beyond microblogging: Conversations and collaboration via Twitter. *In Hawaii International Conference on System Sciences (HICSS)*.
- D. Boyd, S. Golder, G. Lotan. 2010. Retweet: Conversational Aspects of Retweeting on Twitter. *In Hawaii International Conference on System Sciences (HICSS)*.
- D. Kempe. 2003. Maximizing the spread of influence through a social network. *In KDD*.
- D. M. Romero, W. Galuba, S. Asur, B. A. Huberman. 2011. Influence and passivity in social media. *In Proceedings of ACM Conference on World Wide Web (WWW)*.
- E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne. 2008. Finding high-quality content in social media. *In WSDM*.
- E. Bakshy, J. M. Hofman, W. A. Mason, D. J. Watts. 2011. Everyone's an influencer: quantifying influence on Twitter. *In Proceedings of ACM Conference on Web Search and Data Mining (WSDM)*.
- Farahat, A., Nunberg, G., & Chen, F. 2002. Augeas: authoritativeness grading, estimation, and sorting. *In Proceedings of the eleventh international conference on Information and knowledge management*.
- Friedman, J. H. 2001. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*.
- Friedman, J. H. 2006. Recent advances in predictive (machine) learning. *Journal of classification*.
- Institute of Computing Technology, Chinese Lexical Analysis System, <http://ictclas.org/>.
- J. M. Kleinberg. 1998. Authoritative sources in a hyperlinked environment. *In SIAM symposium on Discrete algorithms (SODA)*.
- J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. 2008. Arnetminer: Extraction and mining of academic social networks. *In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD'08)*.
- J. Weng, E. -P. Lim, J. Jiang, Q. He. 2010. TwitterRank: Finding Topic-sensitive Influential Twitterers. *In Proceedings of ACM Conference on Web Search and Data Mining (WSDM)*.
- J. Zhang, M. S. Ackerman, and L. Adamic. 2007. Expertise networks in online communities: structure and algorithms. In WWW.
- L. Page, S. Brin, R. Motwani, and T. Winograd. 1998. The pagerank citation ranking: Bringing order to the web. *Technical report, Stanford Digital Library Technologies Project*.
- M. Bouguessa, B. Dumoulin, and S. Wang. 2008. Identifying authoritative actors in question-answering forums: the case of yahoo! answers. *In KDD*.
- M. Cha, H. Haddadi, F. Benevenuto, K. P. Gummadi. 2010. Measuring User Influence in Twitter: The Million Follower Fallacy. *In Proceedings of AAAI Conference on Weblogs and Social Media (ICWSM)*.
- M. Naaman, J. Boase and C. H. Lai. 2010. Is it Really About Me? Message Content in Social Awareness Streams. *In Computer Supported Cooperative Work*.
- Pal, A., & Counts, S. 2011. Identifying topical authorities in microblogs. *In Proceedings of the fourth ACM international conference on Web search and data mining*.
- Pennacchiotti, M., & Gurumurthy, S. 2011. Investigating topic models for social media user recommendation. *In Proceedings of the 20th international conference companion on World wide web*.
- Pennacchiotti, M., & Popescu, A. M. 2011. Democrats, republicans and starbucks aficionados: user classification in twitter. *In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*.
- P. Jurczyk and E. Agichtein. 2007. Discovering authorities in question answer communities by using link analysis. *In CIKM*.
- Ramage, D., Dumais, S. T., & Liebling, D. J. 2010. Characterizing Microblogs with Topic Models. *In ICWSM*.
- Rao, D., Yarowsky, D., Shreevats, A., & Gupta, M. 2010. Classifying latent user attributes in twitter. *In Proceedings of the 2nd international workshop on Search and mining user-generated contents*.

Comparison between LSA-LDA-Lexical Chains

Costin Chiru, Traian Rebedea and Silvia Ciotec

University Politehnica of Bucharest, Department of Computer Science and Engineering,

313 Splaiul Independetei, Bucharest, Romania

{costin.chiru, traian.rebedea}@cs.pub.ro, silvia.ciotec@gmail.com

Keywords: Latent Semantic Analysis - LSA, Latent Dirichlet Allocation - LDA, Lexical Chains, Semantic Relatedness.

Abstract: This paper presents an analysis of three techniques used for similar tasks, especially related to semantics, in Natural Language Processing (NLP): Latent Semantic Analysis (LSA), Latent Dirichlet Allocation (LDA) and lexical chains. These techniques were evaluated and compared on two different corpora in order to highlight the similarities and differences between them from a semantic analysis viewpoint. The first corpus consisted of four Wikipedia articles on different topics, while the second one consisted of 35 online chat conversations between 4-12 participants debating four imposed topics (forum, chat, blog and wikis). The study focuses on finding similarities and differences between the outcomes of the three methods from a semantic analysis point of view, by computing quantitative factors such as correlations, degree of coverage of the resulting topics, etc. Using corpora from different types of discourse and quantitative factors that are task-independent allows us to prove that although LSA and LDA provide similar results, the results of lexical chaining are not very correlated with neither the ones of LSA or LDA, therefore lexical chains might be used complementary to LSA or LDA when performing semantic analysis for various NLP applications.

1 INTRODUCTION

Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997), Latent Dirichlet Allocation (LDA) (Blei et. al, 2003) and lexical chains (Halliday and Hasan, 1976; Morris and Hirst, 1991) are widely used in NLP applications for similar tasks. All these methods use semantic distances or similarities/relatedness between terms to form topics or chains of words. LSA and LDA use the joint frequency of the co-occurrence of words in different corpora, while the lexical chains technique uses WordNet (<http://wordnet.princeton.edu/>) synsets and links between them to find groups of highly-connected or closely-related words.

Although these methods can be similarly used for various NLP tasks - text summarization (Barzilay and Elhadad, 1997; Gong and Liu, 2001; Haghghi and Vanderwende, 2009), question answering (Novischi and Moldovan, 2006) or topic detection (Carthy, 2004) - they calculate different measures, having different meanings. LDA generates topical threads under a prior Dirichlet distribution, LSA produces a correlation matrix between words and documents, while lexical chains use the WordNet structure to establish a connection between synsets.

Therefore, the comparison and interpretation of similarities and differences between the aforementioned methods is important to understand which model might be the most appropriate for a given scenario (task and discourse type, for example). Previous studies were aimed at comparing different similarity measures built on top of WordNet in order to decide which one gives better results (Barzilay and Elhadad, 1997), or to compare the results provided by the lexical chains built using different measures with the ones given by LSA in order to add a further relationship layer to WordNet for improving its usefulness to NLP tasks (Boyd-Graber et. al, 2006). However, more recently Cramer (2008) pointed out that the existing studies are inconsistent to each other and that human judgments should not be used as a baseline for the evaluation or comparison of different semantic measures.

This work aims to study the behaviour of the three methods: LSA, LDA and lexical chains, based on a series of tests performed on two corpora: one consisting on four Wikipedia articles on different topics and another one built from multi-party online chat conversations debating four pre-imposed topics: forum, chat, blog, wikis.

The paper continues with a review of the

evaluated techniques. Afterwards, we present the procedure for comparing the three methods along with the texts used for evaluation. Section 4 describes the obtained results and our observations, while the last section highlights the main conclusions of the study.

2 EVALUATED METHODS

2.1 LSA – Latent Semantic Analysis

LSA (Landauer and Dumais, 1997) is a statistical method for extracting the relations between words in texts. It is a corpus-based method that does not use dictionaries, semantic networks, grammars, syntactic or morphological parsers, and its input is represented only by raw text divided in “*chunks*”. A chunk may be a sentence, an utterance in a chat, a paragraph or even a whole document, depending on the corpus. The method starts from the term-doc matrix computed on the corpus segmented into chunks and then applies a singular value decomposition in order to compute the most important singular values. Then, it produces a representation in a new space, called the latent semantic space, which uses only the most important (large) k singular values. The value for k depends on the corpus and task, and is usually between 100 and 600, a common choice being 300. This new space is used to compute similarities between different words and even whole documents, practically considering that words that are co-occurring in similar contexts may be considered to be semantically related.

2.2 LDA – Latent Dirichlet Allocation

LDA (Blei et. al, 2003) is a generative probabilistic model designed to extract topics from text. The basic idea behind LDA is that documents are represented as random mixtures of latent topics, where each topic is characterized by a set of pairs word-probability, representing the probability that a word belongs to a topic.

LDA assumes the following generative process for each document in a corpus: for each word $w_{d,i}$ in the corpus, it generates a topic z dependent on the mixture θ associated to the document d and then it generates a word from the topic z . To simplify this basic model, the size of the Dirichlet distribution k (the number of topics z) is assumed to be known and fixed. The Dirichlet prior is used because it has several convenient properties that facilitate inference and parameter estimation algorithms for LDA.

2.3 Lexical Chains

Lexical chains are groups of words that are semantically similar (Halliday and Hasan, 1976; Morris and Hirst, 1991). Each word in the chain is linked to its predecessors through a certain lexical cohesion relationship. Lexical chains require a lexical database or an ontology (most of the time, this database is WordNet) for establishing a semantic similarity between words. For this task, we have used WordNet and the Jiang-Conrath measure (Jiang and Conrath, 1997). As this measure requires the frequency of words in the English language and since we didn't have access to a relevant corpus, we have used the number of hits returned by a Google search for each of the considered words. Once the distances between words were computed, we have used a full-clustering algorithm to group the words in chains. The algorithm worked in an online fashion (each word was evaluated in the order of their appearance in the analyzed text), adding a word to an existing cluster only if it was related to more than 90% of the words that were already part of that chain. If the considered word could not be fitted in any of the existing chains, then we created a new chain containing only that specific word (Chiru, Janca and Rebedea, 2010).

3 COMPARISON METHODOLOGY

Experiments were conducted on two different corpora:

- a corpus composed of four articles from Wikipedia that were debating completely different topics: *graffiti*, *tennis*, *volcano* and *astrology*, consisting of 294 paragraphs and having a vocabulary size of 7744 words. In order not to have our results affected by noise, we removed from the corpus pronouns, articles, prepositions and conjunctions.
- a corpus consisting of 35 online chat conversations debating four pre-imposed topics: forum, chat, blog, wikis, each of them involving between 4 to 12 participants. This corpus consisted of 6000 utterances (41902 words), with a vocabulary size of 2241 words.

3.1 Methods for Obtaining the Results

The SVD is performed using the airhead-research package (<https://code.google.com/p/airhead->

research/wiki/LatentSemanticAnalysis) and a value of $k = 300$. Then, the LSA results are obtained starting from the matrix of similarities between each pair of words in the corpus. The degree of similarity between two words is computed using the cosine of the corresponding vectors in the latent space.

For LDA, the results are obtained from the distribution of each topic's words and the corresponding probabilities. In the first corpus, containing encyclopaedic articles from four different domains, we decided to use a number of topics $k = 4$ for this analysis. For the second corpus, consisting on debates on four imposed topics, we decided to use $k = 5$ topics for the analysis, as besides the imposed topics, the participants also inputted some off-topic content that could have been considered as the fifth topic. In order to better understand the behaviour of LDA, we extracted the top 35, 50, 100, 150 and 200 words that were considered representative for each topic, given that each article contained over 1000 words. The topic models were extracted using MALLET - MACHINE Learning for Language Toolkit (<http://mallet.cs.umass.edu/>).

In the case of lexical chains, we analyzed the words from each chain and also considered the maximum length and the total number of the lexical chains from a document (chat or Wikipedia article).

3.1.1 LDA - LSA Comparison

In order to compare the two methods, we started from the LDA topics and computed an LSA score for each concept from each topic generated by LDA. This score represented the average similarity between the target concept and each of the remaining words from the topic. The assessment of the relationship between LSA and LDA scores distributions was performed using Pearson's correlation coefficient and Spearman's rank correlation coefficient. LSA and LDA have also been compared on several NLP tasks, such as predicting word associations (Griffiths et al., 2007) and automatic essay grading (Kakkonen et al., 2008).

3.1.2 LSA - Lexical Chains Comparison

For comparing these two methods, we determined a similarity value for each lexical chain based on the LSA similarity as follows: we computed the LSA similarity between any pair of two words from the chain and averaged over all the words in that chain. LSA has been previously compared with semantic distances in WordNet (Tsatsaronis et al., 2010), but not with lexical chains.

3.1.3 LDA - Lexical Chains Comparison

This comparison is based on the number of common words between the lexical chains and the LDA topics. For each LDA topic we extracted a number of 35, 50, 100, 150 and 200 words, and computed different statistics for each case. To our knowledge, LDA and lexical chains have only been compared as an alternative for text segmentation (Misra et al., 2009).

4 EXPERIMENTAL RESULTS

4.1 Wikipedia Corpus

4.1.1 LDA - LSA Comparison

Table 1 presents the top 10 words from the 4 LDA topics of the first corpus. In Table 2 we present the most similar 30 word-pairs generated by LSA. We need to mention that LSA was trained on the concatenation of all 4 articles from Wikipedia.

Table 1: Top 10 words from the LDA topics for the Wikipedia corpus.

| Topic 0 | Topic 1 | Topic 2 | Topic 3 |
|----------|---------|-----------|-------------|
| graffiti | tennis | volcanoes | astrology |
| new | game | volcano | been |
| culture | player | lava | Chinese |
| form | first | volcanic | personality |
| york | players | surface | scientific |
| design | two | example | based |
| popular | court | formed | considered |
| hip | three | examples | birth |
| style | point | extinct | bce |
| spray | French | flows | belief |

Table 2: Top 30 most similar word-pairs generated by LSA for the Wikipedia corpus.

| LSA Word Pairs | | |
|----------------|----------------|-------------------------|
| men-cup | mid-thinning | plates-tectonic |
| mid-crust | tie-addition | center-baseline |
| hop-music | thinning-ridge | choice-receiver |
| mid-ridge | pace-receiver | depicted-dealer |
| lake-park | shift-equinox | degrees-equinox |
| mm-bounce | basque-perera | gladiatorial-cil |
| lady-week | degrees-shift | difficult-extinct |
| são-brazil | rhode-newport | tectonic-ridge |
| force-hero | federation-itf | era-compete |
| test-results | mud-formation | lifespans-volcanologist |

For each topic we plotted the distributions of LDA and LSA scores for each word from that topic,

computed as described in the previous section. Each LDA topic has 35 words that are sorted decreasing according to the LSA scores. The best result we have obtained was for the Topic 1 (*tennis*), where with very few exceptions, the LSA and LDA scores were very well correlated (0.855). This case is presented in Figure 1, where the x-axis represents the word number from the LDA topic and on the y axis we plotted the LDA and LSA scores corresponding to that word. The words' probabilities for the considered topic computed with LDA are represented by the blue colour while in red we present the LSA scores. The scattering diagram for the same topic is presented in Figure 2.

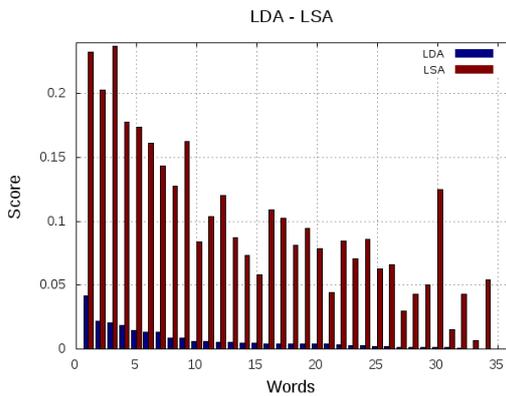


Figure 1: LDA – LSA distributions for Topic 1 (*tennis*) from the Wikipedia corpus.

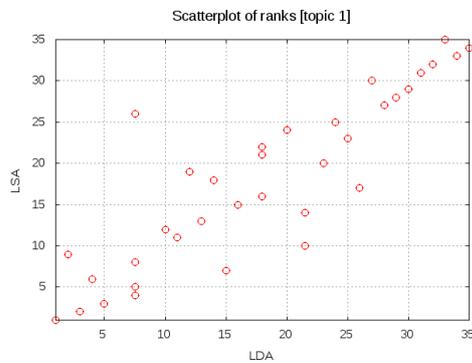


Figure 2: Scattering plot for the rank distributions for the LDA – LSA comparison for Topic 1 (*tennis*).

For a better visualization of the relationship between the two distributions, we present in Table 3 the Pearson's correlation and the Spearman's rank correlation coefficients between the LDA and LSA scores for each of the four LDA topics. With one exception, these values are close to 1, indicating a very good correlation (the strongest is highlighted in bold).

Table 3: LDA-LSA Pearson's Coefficient for the Wikipedia corpus.

| Topic | Pearson's Coefficient | Spearman's Coefficient |
|---------------------|-----------------------|------------------------|
| 0 (graffiti) | 0.560 | 0.778 |
| 1 (tennis) | 0.855 | 0.873 |
| 2 (volcanoes) | 0.782 | 0.840 |
| 3 (astrology) | 0.745 | 0.745 |

These results prove that there is clearly a correlation between the two distributions because both tend to decrease towards the last words of the topic. However, there are some words for which the two scores are discordant. We have extracted them and obtained the following results:

- for Topic 0 (graffiti): hip, produced, styles, non, offered, property;
- for Topic 1 (*tennis*): point, receiving;
- for Topic 2 (volcanoes): extinct, gases, features, falls;
- for Topic 3 (astrology): considered, challenge, avoid.

It is interesting to observe that the better correlated the LSA and LDA scores are for a given topic, the more the words underestimated by LSA correspond to that topic.

4.1.2 LSA - Lexical Chains Comparison

Using the LSA similarity between words, we computed a score ranging from 0 to 1 for every lexical chain.

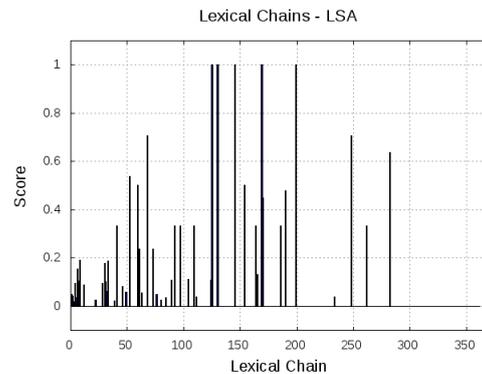


Figure 3: LSA scores for the lexical chains of the *tennis* article from the Wikipedia corpus.

For an example of the obtained results, see Figure 3 (for Topic 1 - *tennis*) where on the x-axis are the lexical chains (excluding those formed only by one word) and on the y-axis are their LSA scores.

We have noticed that the best lexical chains are obtained for the texts that had also a good

correlation between the scores obtained by LDA and LSA. Also, one can see that there are only few lexical chains which are closely related in terms of LSA, which leads us to believe that LSA and lexical chains are not very well correlated.

Approximately 70% of the generated lexical chains were composed of a single word. In the rest of the lexical chains, the most frequent ones are those having small LSA scores - in the range (0, 0.25]. The other intervals represent only a small percent from the number of chains remaining when the single word chains are ignored.

The LSA scores are dependent on the lexical chain length, so we considered that it would be interesting to draw a parallel between these two elements. In Figure 4 are plotted the lexical chains lengths with their corresponding LSA scores for the tennis article. The x-axis contains the lexical chains indexes and the z-axis contains the LSA score and the length of that chain.

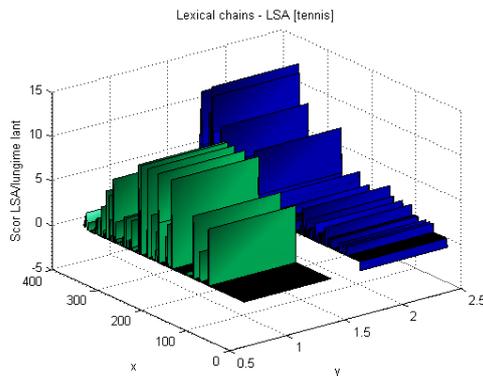


Figure 4: The LSA scores (green) and the lexical chains length (blue) from the tennis article.

4.1.3 LDA - Lexical Chains Comparison

For this comparison, we generated the most representative words for each of the four topics keeping the top 35, 50, 100 and 200 words and gradually comparing the number of common words between the topics and the lexical chains. It should be mentioned that a word can be representative for multiple topics (having different probabilities for each topic). The maximum lengths of the lexical chains from each article were 31, 28, 24 and 12 words for the articles about volcanoes, graffiti, astrology and tennis respectively. In the case of LDA topics having 35 words, the common words between LDA and lexical chains were:

- *Volcano* article: volcano, lava, surface, example, extinct, flow, explosive, water,

generally, volcanism, fire, form, fluid, field, few, weight, first;

- *Tennis* article: tennis, game, player, first, court, point, french, receiver, real, playing, wide, cup, usually, full, current, covered, recent;
- *Graffiti* article: graffiti, new, culture, form, york, design, popular, hip, style, spray, paint, early, different, day, rock, history, elements, stencil, due, chicago, dragon, disagreement, newspaper, egypt, popularity, production;
- *Astrology* article: astrology, chinese, personality, scientific, birth, belief, challenge, astronomical, astronomy, avoid, philosophy, babylonian, basis, basic, average, birthday, beginning, century, believe.

In order to compare the results between LDA and lexical chains, we determined how many chains contained words that were also considered representative for the four LDA topics along with the number of such common words.

First of all, we computed for each topic the first 35 words and represented the frequency of common words between the lexical chains and the topics of this size. In this case, most chains had no common words with any of the topics (more than 700 such chains). The Topic 0 (*graffiti*) had one common word with the largest number of lexical chains (over 25 chains), the Topic 1 (*tennis*) had a common word with 17 such chains, while the last topic (*volcano*) had words in 15 lexical chains. Topic 2 (*astrology*) had two common words with 3 lexical chains (most chains comparing with the other topics), but had a smaller number of lexical chains (13) with which it had a single word in common. As an overall statistic, the words from Topic 0 (*graffiti*) could be found in the most lexical chains. After we increased the number of words to 50 per topic, around 430 chains had no word in common with the topics, and the number of most common words between topics and lexical chains increased to 3, although there were only two such chains – one for Topic 1 (*tennis*) and one for Topic 3 (*volcano*). Further increasing the number of words in a topic to 100, we saw that Topic 3 (*volcano*) had 4 common words with one lexical chain and, compared to the previous case, this time all the topics have found 3 common words with at least one lexical chains. At this point, Topic 1 (*tennis*) had a single word in common with over 40 lexical chain, this becoming the best score, comparing to the previous cases when the Topic 0 (*graffiti*) was the most popular in this category. Overall, the Topic 3's words are the most often found in the lexical chains (over 40 chains having o

word in common, 2 having 2 words in common and 1 with 3 and 1 with 4 words in common).

Finally we increased the number of words per topic to 200 (Figure 5). Also in this case, there still remained around 350 chains that had no words in common with any of the topics. It can be seen that the Topic 3 (*volcano*) has 7 words in common with one of the lexical chains (the best score so far), while Topic 2 (*astrology*) had 5 common words with one of the chains. The details of this discussion are summarized in Table 4.

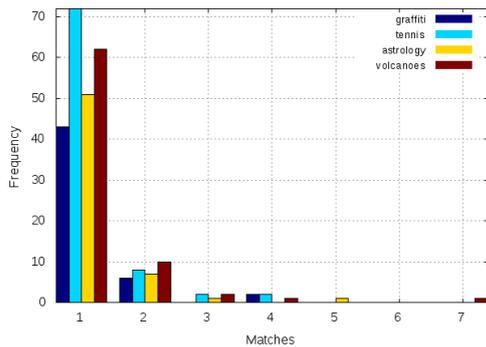


Figure 5: The distribution of the common words between topics (of 200 words) and the lexical chains.

Table 4: Number of chains having a single word in common with different topics (highest values are in bold), and the maximum number of words in common with a topic in a single chain.

| Topic words/ topic | T0 | T1 | T2 | T3 | No topic | Max. common words |
|--------------------|-----------|---------------|-----|---------------|----------|-------------------------------------|
| 35 | >25 | 16 | 12 | 15 | >300 | 2 (3 chains for T2, 1 for the rest) |
| 50 | 29 | 17 | 15 | 20 | ~300 | 3 (T1 & T3) |
| 100 | 24 | >40 | 33 | >40 | ~270 | 4 (T3) |
| 150 | 34 | 51 | 41 | >50 | ~260 | 6 (T3) |
| 200 | >40 | >70 | >50 | >60 | ~250 | 7 (T3) |

In conclusion, the most frequent situation (besides the lexical chains having no word in common with the topics) is the one when the lexical chains and the topics have exactly one common word, and the maximum number of common words that was found was 7 for topics consisting of 200 words.

4.2 Chat Conversations Corpus

A similar methodology was used to compare the results on the chat corpus in order to see if there are any noticeable differences due to the change of the type of discourse. The results are reported more

briefly in this section.

4.2.1 LDA - LSA Comparison

Table 5 presents the top 10 words from the 5 LDA topics. In Table 6 we present the most similar 30 word-pairs generated by LSA.

Similarly to the Wikipedia corpus, we plotted the distributions of LDA and LSA scores for each word from that topic and obtained the best result for Topic 1 (0.73). This case is presented in Figure 6, while in Figure 7 we present the scattering diagram for this topic. The Pearson's and the Spearman's Rank correlation coefficient between the LDA and LSA scores for each LDA topics are presented in Table 7.

Table 5: Top 10 words from the LDA topics in the chat corpus.

| Topic 0 | Topic 1 | Topic 2 | Topic 3 | Topic 4 |
|----------|-----------|----------------|-------------|---------|
| Forums | wiki | blogs | chat | blog |
| Internet | solutions | brain storming | information | person |
| Good | solve | company | friends | forum |
| Ideas | opinion | clients | find | board |
| Right | web | changes | folksonomy | certain |
| Users | wave | compare | follow | fun |
| write | number | cases | great | new |
| idea | need | different | hard | part |
| people | like | easy | integrate | change |
| help | use | more | maybe | friend |

Table 6: Top 30 most similar word-pairs generated by LSA in the chat corpus.

| LSA Word Pairs | | |
|---------------------|------------------------------|-------------------------|
| traveller-messaging | patterns-vmtstudents | mathematicians-patterns |
| sets-colinear | flame-wars | dictate-behaviour |
| decides-improper | physically-online | satisfaction-conducted |
| easily-switchboard | inconvenient-counterargument | counts-popularity |
| ads-revenue | induction-patterns | editors-objectivity |
| supplying-focuses | inconvenient-counter | duties-minimum |
| patient-recall | sets-colinear | decides-improper |
| hm-conversions | equations-quicksilver | lie-proud |
| secure-hacked | simplifies-equals | chatroom-leaves |
| careful-possible | fellow-worker | hexagonal-array |

As it was expected, the results for the chat corpus are less correlated than the ones obtained for the Wikipedia corpus. This drop in performance can be partly explained by the increased number of topics (one additional topic), but mostly by the different nature of discourse: the Wikipedia articles are much

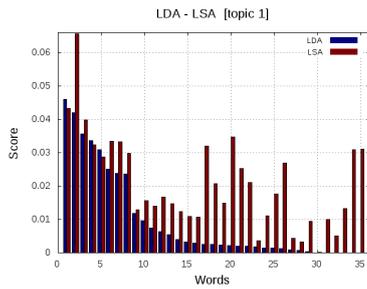


Figure 6: LDA – LSA distributions for Topic 1 from the chat corpus.

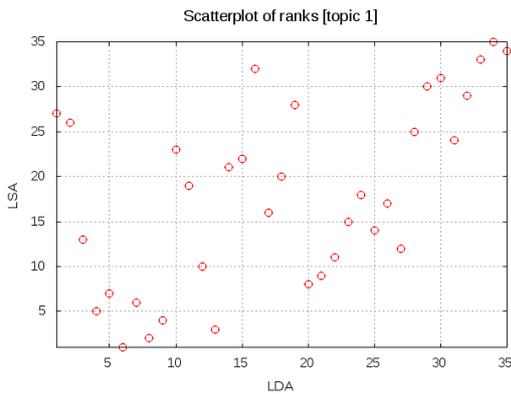


Figure 7: Scattering plot for the ranks distributions for the LDA–LSA comparison for Topic 1 from the chat corpus.

Table 7: LDA-LSA Pearson’s Coefficient for the chat corpus.

| Topic | Pearson’s Coefficient | Spearman’s Coefficient |
|-------|-----------------------|------------------------|
| 0 | 0.63 | 0.46 |
| 1 | 0.73 | 0.55 |
| 2 | 0.55 | 0.41 |
| 3 | 0.46 | 0.35 |
| 4 | 0.71 | 0.32 |

more focused/cohesive and coherent than chat conversation between multiple participants. It also provides an insight related to the content of the chat conversations: it seemed that the topic one (related to *wikis/Wikipedia*) discovered by LDA was more coherent than the other topics, at least by looking at the LSA correlation scores. The second highest score in this hierarchy was for the *forum-blog* topic showing that the participants do not perceive significant differences between these concepts. However, the most intriguing result was the placement of the third topic (related to *chat*) on the last place, showing the least coherence. We expected that this topic to have in fact the highest coherence, being the tool most frequently used by the participants and therefore the tool that they knew

best. These results may also be influenced by the way we are measuring the coherence of a LDA topic through its correlation with the average LSA similarity scores.

4.2.2 LSA - Lexical Chains Comparison

For the chat corpus, the values of the LSA similarity between words for every lexical chain ranged from -1 to 1, as it can be seen in Figure 8. We can observe that the correlation between the LSA and lexical chains for the chat corpus is lower than the one for the Wikipedia corpus, this fact being generated by the lower cohesion of the text in this case.

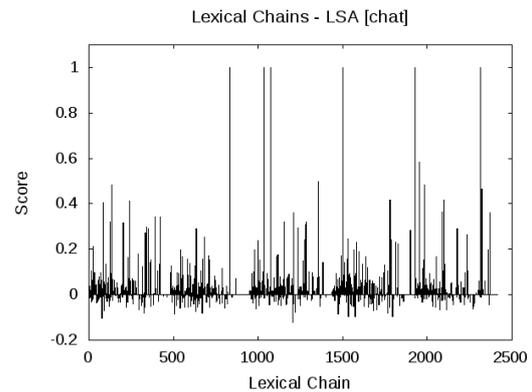


Figure 8: LSA scores for the lexical chains from the chat corpus.

4.2.3 LDA - Lexical Chains Comparison

Similarly to the Wikipedia corpus, each of the five topics was generated keeping the top 35, 50, 100 and 200 words and gradually comparing the number of common words between the topics and the lexical chains. The maximum length of the lexical chains from this corpus was 84, much larger than the one obtained in the case of the Wikipedia corpus. This is due to the fact that the four topics imposed for debating in the chat conversations (*forum*, *chat*, *blog*, and *wikipedia*) were strongly related compared to the Wikipedia articles that debated topics from different domains.

The number of common words is predominantly 1, reaching a maximum of 8 common words for the third topic (related to *chat*) for a length of the lexical chain of 150 words. The results are similar to those obtained for the Wikipedia corpus.

5 CONCLUSIONS

In this paper we discussed the characteristics and behaviour of three methods frequently used to assess semantics in various NLP applications: LSA, LDA and lexical chaining. These methods have been tested on two different corpora containing different types of written discourse: a corpus consisting of 4 articles from Wikipedia and another one consisting of 35 chat conversations with multiple participants debating four pre-imposed topics: forum, chat, blog and wikis.

In contrast with the previous studies, we have compared the outcomes of the three methods using quantitative scores computed based on the outputs of each method. These scores included correlations between similarity scores and the number of common words from topics and chains. Thus, the obtained results are task and discourse-independent.

The most important result is that LSA and LDA have shown the strongest correlation on both corpora. This is consistent with the theoretical underpinnings, as LDA is similar to Probabilistic Latent Semantic Analysis (pLSA), except that the LDA distribution of topics is assumed to have a prior Dirichlet distribution. Moreover, LSA scores might be used to compute the coherence of a LDA topic as shown in the paper.

Another important contribution is that WordNet-based lexical chains are not very correlated with neither LSA nor LDA, therefore they might be seen as complementary to the LSA or LDA results.

ACKNOWLEDGEMENTS

This research was supported by project No.264207, ERIC-Empowering Romanian Research on Intelligent Information Technologies/FP7-REGPOT-2010-1.

REFERENCES

- Barzilay, R. and Elhadad, M., 1997. Using lexical chains for text summarization. In: *Proceedings of the Intelligent Scalable Text Summarization Workshop*, pp. 10–17.
- Budanitsky, A. and Hirst, G., 2006. Evaluating wordnet-based measures of semantic relatedness. In: *Computational Linguistics 32 (1)*, pp. 13–47.
- Blei, D. M., Ng, A. Y. and Jordan, M. I., 2003. Latent Dirichlet allocation. In: *Journal of Machine Learning Research 3*, pp. 993–1022.
- Boyd-Graber, J., Fellbaum, C., Osherson, D. and Schapire, R., 2006. Adding dense, weighted, connections to WordNet. In: *Proceedings of the 3rd GlobalWordNet Meeting*, pp. 29–35.
- Carthy, J., 2004. Lexical chains versus keywords for topic tracking. In: *Computational Linguistics and Intelligent Text Processing, LNCS*, pp. 507–510. Springer.
- Chiru, C., Janca, A., Rebedea, T., 2010. Disambiguation and Lexical Chains Construction Using WordNet. In S. Trăușan-Matu, P. Dessus (Eds.) *Natural Language Processing in Support of Learning: Metrics, Feedback and Connectivity, MatrixRom*, pp. 65–71.
- Cramer, I., 2008. How well do semantic relatedness measures perform? a meta-study. In: *Proceedings of the Symposium on Semantics in Systems for Text Processing*.
- Griffiths, T. L., Steyvers, M. and Tenenbaum, J. B., 2007. Topics in semantic representation. In: *Psychological Review*, vol. 114, no. 2, pp. 211–244.
- Gong, Y. and Liu, X., 2001. Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In: *Proceedings of the 24th ACM SIGIR conference*, pp. 19–25.
- Haghighi, A. and Vanderwende, L., 2009. Exploring content models for multi-document summarization. In: *Proceedings of HLT-NAACL*, pp. 362–370.
- Halliday, M. A.K. and Hasan, R., 1976. *Cohesion In English*, Longman.
- Jiang, J. J. and Conrath, D. W., 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In: *Proceedings of ROCLING X*, pp. 19–33.
- Kakkonen, T., Myller, N., Sutinen, E. and Timonen, J., 2008. Comparison of Dimension Reduction Methods for Automated Essay Grading. In: *Educational Technology & Society*, 11(3), pp. 275–288.
- Landauer, T. K. and Dumais, S. T., 1997. A solution to Plato's problem: the Latent Semantic Analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2), 211–240.
- Misra, H., Yvon, F., Jose, J. and Cappé, O., 2009. Text Segmentation via Topic Modeling: An Analytical Study. In: *18th ACM Conference on Information and Knowledge Management*, pp. 1553–1556.
- Morris, J. and Hirst, G., 1991. Lexical Cohesion, the Thesaurus, and the Structure of Text. In: *Computational Linguistics*, Vol 17(1), pp. 211–232.
- Novischi, A. and Moldovan, D., 2006. Question answering with lexical chains propagating verb arguments. In: *Proceedings of the 21st International Conference on CL and 44th Annual Meeting of ACL*, pp. 897–904.
- Tsatsaronis, G., Varlamis, I. and Vazirgiannis, M., 2010. Text relatedness based on a word thesaurus. In: *Artificial Intelligence Research*, 37, pp. 1–39.

Prediction of Human Personality Traits From Annotation Activities

Nizar Omheni¹, Omar Mazhoud¹, Anis Kalboussi¹ and Ahmed Hadjkacem²

¹Higher Institute of Computer and Management, University of Kairouan, ReDCAD Laboratory,
Khmais Alouini Street 3100, Kairouan, Tunisia

²Faculty of Economics and Management, University of Sfax, ReDCAD Laboratory,
Road of the Airport Km4 3018, Sfax, Tunisia

{omheninizar, omarmazhoud, kalboussianis}@gmail.com, ahmed.hadjkacem@fsegs.rnu.tn

Keywords: Annotation, Personality, Big Five Personality Model.

Abstract: We show how reader's annotation activity captured during an active reading session relates to their personality, as measured by the standard Five Factor Model. For 120 volunteers having usually the habit of reading, we gather personality data and annotation practices. We examine correlations between readers personality and such features of their annotative activities such as the total number of annotation acts, average number of annotation acts, number of textual annotation acts, number of graphical annotation acts, number of referential annotation acts and number of compounding annotation acts. Our results show significant relationships between personality traits and such features of annotation practices. Then we show how multivariate regression allows prediction of the readers personalities traits given their annotation activities.

1 INTRODUCTION

Studying human activity and interaction with technology has grown dramatically over the last decade. Yet studying reading poses particular challenges. (Marshall,2010) reported the citation of Tzvetan Todorov, quoted by Nicholas Howe in Jonathan Boyarins compilation, the *Ethnography of Reading*: "Nothing is more commonplace than the reading experience, and yet nothing is more unknown. Reading is such a matter of course that at first glance it seems there is nothing to say about it". Although details of reading activity (moving eyes, writing annotation...) may tell us something about the reader. When people read and interact actively with their reading materials they do unselfconscious activities which can be keys features to their personalities.

For decades, the psychologists search to understand the human personality and to find a systematic way to measure it. After several researches they show a relation of dependence between human personality traits and different behaviors. (Ryckman,2010) reported the Allport's ¹ definition of personality: "personality is the dynamic organization within the in-

¹Gordon Willard Allport (November 11, 1897 October 9, 1967) was an American psychologist. He was one of the first psychologists to focus on the study of the personality, and is often referred to as one of the founding figures of personality psychology.

dividual of those psychophysical systems that determine his characteristic behavior and thought". Thus, in Allports view human behavior is really controlled by internal forces known as the personality traits.

This paper attempts to bridge the gap between reading activity research and personality research through reader's annotation practices. Our core research question asks whether annotation activity can predict personality traits. If so, then there is an opportunity to use a natural human practice as a new source to better understand the reader personality.

Several works has shown the opportunity of predicting user personality using the information people reveal in their online social profile (Twitter, Facebook) (Bachrach et al,2012) (Golbeck et al,2011). They refer to what people share, self-description, status updates, photos, tags, etc. We pretend that annotative activity is more spontaneously and natural practice and it can reveal something about human personality.

Personalization attracted increased attention in many areas. So the need to predict personality traits increases over time mainly when several research has shown the link between personality traits and success in human relationship and practices (Barrick and Mount,1991) (Eswaran et al,2011). By nature an introverted person is not interested to make so much relation with other while an extravert person do. Actually, certain developed recommendation systems con-

sider the personality traits as key feature of recommendation (Nunes et al,2008).

The paper is structured as follows: In Section 2, we present background on the Big Five personality index. Then we present our experimental setup and methods. In the third section we present the results on correlation between each annotative activity feature and personality factor. Next, we show how multivariate regression allows prediction of annotators personalities traits given their annotation activities. We conclude with a discussion of the possible implications that this work has for such domains of application.

2 BACKGROUND AND RELATED WORK

2.1 The Big Five Personality Model

The big five personality traits are the best accepted and most commonly used scientific measure of personality and have been extensively researched (Peabody and De Raad,2002). That personality is well described as five traits was discovered through the study of the adjectives from natural language that people used to describe themselves and then analyzing the data with a statistical procedure known as factor analysis that is used to reduce lots of information down to its most important parts. In the following we cite a brief explanation of the five personality traits.

2.1.1 Openness to Experience

Openness includes traits like imagination, appreciation for art, depth of emotions, adventure, unusual ideas, intellectual curiosity, and willingness to experiment. People who score high in openness like usually to learn new things and enjoy new experiences.

2.1.2 Conscientiousness

Conscientiousness includes traits like orderliness, selfdiscipline, deliberateness, and striving the achievement. People that have a high degree of conscientiousness are planned, have the tendency to act dutifully, have the sense of responsibility and competence.

2.1.3 Extraversion

Extraversion includes traits like energy, positive emotions, surgency, assertiveness, sociability and talkativeness. Extraverts people get their energy from

interacting with others, while introverts get their energy from within themselves.

2.1.4 Agreeableness

Agreeableness includes traits like trust in others, sincerity, altruism, compliance, modesty and sympathy. People that have high degree of agreeableness are friendly, cooperative, and compassionate, while people with low agreeableness may be more distant.

2.1.5 Neuroticism

Neuroticism relates to ones emotional stability and degree of negative emotions. This dimension measures the people degree of anxiety, angry, moodiness, and the sensitivity to stress. People that score high on neuroticism often experience emotional instability and negative emotions.

2.2 Related Work

In (Burger,2011) view the personality is a "consistent behavior patterns and intrapersonal processes originating within the individual". Trait psychologists assume that personality is relatively stable and predictable (Burger,2011). So, several research work has been done with personality traits as it influences human decision making process and interests. (Nunes et al,2008) pioneered the model and implement of personality traits in computers. Indeed, (Nunes et al,2008) propose to model the user's traits in a profile which they called User Psychological Profile - UPP. In order to fill in the profile UPP the authors utilised an online tool called the NEO-IPIP² inventory based on 300 items. Through user's answers to NEO-IPIP inventory the authors are able to predict the user personality. Through their experimentation (Nunes et al,2008) try to prove that Recommender Systems can be more efficient if they use the User Psychological Profile (UPP). Although the authors follow an explicit way to predict the user traits, the results presented in (Nunes,2008) are fruitful. (Tkalcic et al,2009) propose a personality-based approach that is based on the big five model for collaborative filtering Recommender Systems. In fact, the authors calculate the user personality scores by means of a questionnaire. Then they measure the user similarity, that is based on personality, that yields a list of close neighbours. This list is used after as a database to compile a personalized list of recommended items.

²The NEO-IPIP is a computer based Personality Inventory, able to measure people Personality Traits created by John Johnson (Johnson.).

(Roshchina et al,2011) propose personality-based recommender system which the aim is to select for the user reviews that have been written by like-minded individuals. The user similarity is calculated based on the personality traits according to the Big Five model. The authors predict the personality traits based on linguistic cues collected from the user-generated text. (Bachrach et al,2012) and (Golbeck et al,2011) show the relationship between personality traits and various features of social media (FaceBook, Twitter). Their findings prove the possibility to predict accurately the user's personality through the publicly available information on their social network profile.

As it is mentioned above most of works has been done with the "Big Five" model of personality dimensions which has emerged as the most well-researched and regarded measures of personality in last decades. The best of our knowledge, our work is among the first to look at the relationship between annotation activity and personality traits. Much works try to predict personality from what the user offer consciously (answers to a questionnaire, informations available on public social profile...). Despite the fact that the findings of these researchs are fruitful we believe that predicting personality from annotative activity is more credible as the annotation is defined as "a basic and often unselfconscious way in which readers interact with texts" (Marshall,2010).

Due to the spontaneously and unselfconscious aspects of annotation we are interested to predict personality from this potential source of knowledge.

3 DATA COLLECTION

We consider group of 120 volunteers. The subjects selected were recruited with respect to certain criterias. Infact, the age of our volunteers should be between 18 and more and they should be academic people. In our sample we have the two sex (44 women and 76 men). Another criteria for selection , we asked if the volunteer has the habit of reading and does he annotated his documents frequently. If all these conditions exist the subject can be selected to our experimentation.

Each subject was instructed to answer a standard Five Factor Model questionnaire (the NEO-IPIP Inventory). Then, he obtained a feedback regarding his personality based on his responses. This step gives us the personality scores based on the Big Five Model for each volunteer. To associate personality scores to subjects annotative activities, we gather annotation practices for each people. Here, we collect documents annotated in a spontaneous and natural way. So we asked, first of all, if the subject had a document an-

notated previously (academic course, book,...). If not we asked him what topics interest him, then we give him an article with few pages to not weary him.

We are very careful to the comfortability of the volunteers during the experience to guarantee their spontaneous and natural reactions. Thus they are free to choose places and conditions to read and annotate the documents and they have enough time to do. The strategy followed give us fruitful results. Infact, the different subjects (who have not a document annotated previously) interact actively with the reading materials in view of the feel of comfortableness and the interest to the document read.

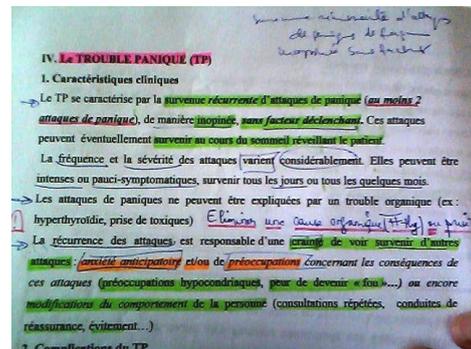


Figure 1: Annotation practices of a reader.

3.1 User Annotation Activity

(Marshall,2010) defines the annotation as "a basic and often unselfconscious way in which readers interact with texts". We mean by annotation the act to add a critical or explanatory notes to a written work, to highlight a passage, to write down, and so on marks the reader makes on a page during his reading activity. To fulfil our experimentation, we ask each subject to give us an annotated paper document. Then, we analyse the readers annotations to extract some features. We started by classifying annotations in the three general categories cited by (Agosti and Ferro,2003) : graphical annotation acts, textual annotation acts, and reference annotation act all depends to the materialization sign of the semantics of the annotation added to the annotated document. Then, for each reader, we collect a simple set of statistics about their annotative activity. These included the following:

1. Total Number of Annotation Act (TNAA)
2. Average Number of Annotation Act (number of annotation acts per a single annotated page)(ANAN)
3. Number of Graphical Annotation Act (NGAA)
4. Number of Textual Annotation Act (NTAA)

5. Number of Reference Annotation Act (NRAA)
6. Number of compounding Annotation Act (textual sign, graphic sign and reference sign of annotation act can be compounded together in order to express complex meanings of annotation)(NCAA).

This set of statistics tends to characterize quantitatively the reader’s annotation practices. Next we run a Pearson correlation³ analysis between subjects’ personality scores and each of the features obtained from analyzing their annotative activities.

4 PERSONALITY AND ANNOTATION FEATURES CORRELATION

We study the Pearson correlation between subjects’ personality scores and each of the features obtained from analyzing their annotative activities. We report the correlation values in table I. Those that were statistically significant for $p < 0.05$ are bolded.

Table 1: Pearson correlation values between annotation features scores and presonality scores.

| | Open. | Consc. | Extra. | Agree. | Neuro. |
|------|--------|--------------|---------------|--------|---------------|
| TNAA | -0,059 | 0,128 | -0,138 | 0,089 | -0,287 |
| ANAA | 0,003 | 0,080 | -0,210 | 0,163 | -0,183 |
| NGAA | -0,067 | 0,040 | -0,130 | 0,105 | -0,207 |
| NTAA | 0,001 | 0,182 | 0,040 | 0,085 | -0,211 |
| NRAA | -0,075 | 0,045 | -0,122 | 0,077 | -0,207 |
| NCAA | -0,059 | -0,012 | -0,147 | 0,014 | -0,219 |

We found in our analysis fewer significant correlations, but we believe a larger sample size would produce much better results. However, the results we obtained even with a small sample show promise that the annotative activity can be useful for computing such personality traits. In fact, table I shows significant correlations for Neuroticism, Conscientiousness, and Extraversion traits. We need larger sample to verify the inference of the other traits from peoples annotations.

Next, we present the scatter plots for the most significant correlations between annotation practices features and personality traits. These plots presenting the relationship between annotative activity features and human traits, where horizontal axis represents the average personality trait scores and the vertical axis represnts the annotative activity feature values.

³Pearson’s correlation $r \in [-1, 1]$ measures the linear relationship between two random variables.

4.1 Conscientiousness

As presented in table I Conscientiousness is positively related to the number of textual annotation act (fig.2). The rest of the correlation values are not considered because of p -value > 0.05 . But this is not a reason to reject definitively the rest of annotation features as a larger sample size may produces other significant correlations.

The considered correlation may indicates that conscientious people are interested to use textual annotation acts. Infact, conscientious individuals are prudent which means both wise and cautious, better organized and they avoid acting spontaneously and impulsively. Thus, it may be the case that people who have high degree of conscientiousness are interested to use textual annotation more than other annotation acts as it demands more reflexion, reasoning and cognitive effort.

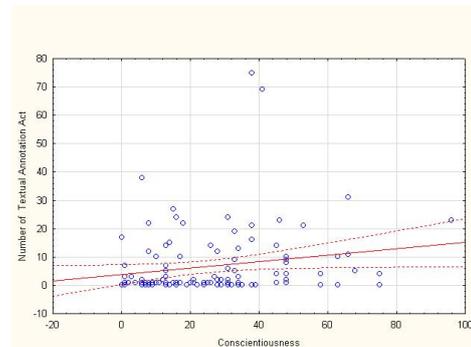


Figure 2: Scatter Plot showing Number of Textual Annotation Act against Conscientiousness scores.

4.2 Extraversion

According to results shown in table I, Extraversion is negatively correlated with the average number of annotation act (fig.3). The rest of the correlation values can be probably significant with a larger sample size. We can interpret the regression fit shown in figure 3 as follow: The fit is correlated negatively which is not surprising as extraversion is marked by pronounced engagement with the external world where extraverts tend to be energetic and talkative while introverts are more likely to be solitary and reserved. Thus, it may be the case that reading and annotation is an intimate activities, we do it in private, so people who are socially active are less willing to practice annotation.

4.3 Neuroticism

Table I shows that neuroticism is negatively correlated with all the features of annotation activity. Here,

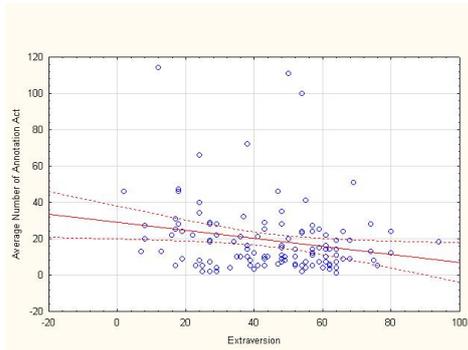


Figure 3: Scatter Plot showing Average Number of Annotation Act against Extraversion scores.

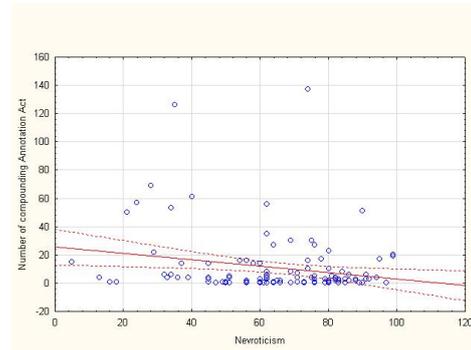


Figure 5: Scatter Plot showing Number of compounding Annotation Act against Neuroticism scores.

the chosen sample size is sufficient to have significant correlations for all the annotation features. The different correlation values are very significant which can show the sensitivity of annotation practices to the neuroticism trait.

In other hand, one possible explanation for these correlations is that more Neurotic people are emotionally reactive and they experience negative emotions for unusually long periods of time which can diminish the neurotics ability to think clearly and make decisions. Thus those who score high on Neuroticism are less eager to use annotation act as they can not actively and critically engaging with the content for a long periods of time.

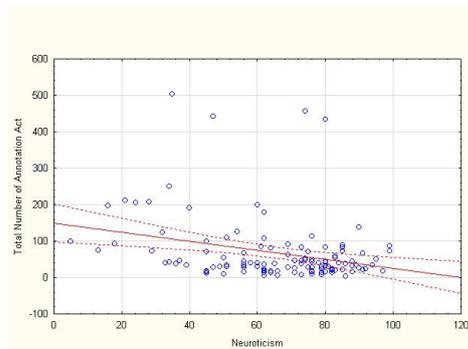


Figure 4: Scatter Plot showing Total Number of Annotation Act against Neuroticism scores.

4.4 Openness and Agreeableness

Unfortunately, the correlation values related to the Openness and Agreeableness traits are very low. But we can not reject definitely the hypothes of prediction of these traits from annotation activity. We may obtain significant values if we increase the sample size. As an example, the p -value of the regression fit of the Average Number of Annotation Act against the Agreeableness traits is $p = 0.076$. This value can be ameliorated with a larger sample size.

5 PREDICTING PERSONALITY

Previously we examined the correlations between each of Big Five personality dimension and annotative activity features. Now, we are interested to make predictions about a subject’s personality based on multiple annotation features. First of all, we used the multivariate linear regression to predict each personality trait using the annotation features. Next, we used the coefficient of multiple determination R^2 to measure the strength of fit. Also, we measure the F-test to verify the statistical significance of the collective influence that have the annotation features on the personality traits. Thus, larger values of the F-test statistic provide stronger evidence against H_0 ⁴. To reject H_0 the value of the F-test should exceeds a critical value calculated as follow:

$$F = \frac{R^2/k}{(1 - R^2)/[n - (K + 1)]}$$

Where k is the number of explanatory variables in our model which corresponds to the number of annotation activity features ($K=6$) and n represents the sample size ($n=120$). So the $F_{observed}$ is compared against a critical F with 6 degree of freedom in the numerator and $n-7$ degrees of freedom for error in denominator. In our case $F_{critical}=2.18$ for alpha level⁵ of 0.05. Results shown in table II indicate that the null hypothesis is rejected for two cases. In fact, Neuroticism and Conscientiousness can be predicted with reasonable accuracy using features of annotation activity, whereas other traits are more difficult to be predicted using annotations. Prediction regarding Conscientiousness is reasonably accurate, with R^2 value of 0.12, $F_{observed}$

⁴The null hypothesis states that there is no relationship between annotation activity features and personality traits.

⁵The alpha level is defined as the probability of what is called a Type I error in statistics. That is the probability of rejecting H_0 when in fact it was true.

value of 2.52 which exceeds the $F_{critical}$ value and P-value of 0.03 which is lower than the α value where P-value is the probability the F-test statistic is larger than the observed F-value. For Neuroticism we obtained the model with the best fit, with an R^2 value of 0.14, $F_{observed}$ value of 3.11 and P-value of 0.01, indicating quite accurate a prediction. The model for Extraversion has a lower fit and the model for Agreeableness is even less accurate. It seems that Openness is the hardest trait to predict using annotation activity features.

Table 2: Predicting personality traits using annotation activity features through multivariate linear regression.

| Trait | R^2 | F test | P-value |
|-------------------|-------|--------|---------|
| Openness | 0.03 | 0.57 | 0.76 |
| Conscientiousness | 0.12 | 2.52 | 0.03 |
| Extraversion | 0.07 | 1.32 | 0.25 |
| Agreeableness | 0.05 | 1.03 | 0.41 |
| Neuroticism | 0.14 | 3.11 | 0.01 |

6 DISCUSSION

In this study we show that Neuroticism and Conscientiousness traits are correlated with annotation activity features. We expect a larger sample size can be helpful to verify the correlation of the other human traits to annotation practices.

Our findings are based on pen-and-paper approach which is qualified by its relative ease with which the reader may interact with a document in an intuitive and familiar manner.

Recent researchs endeavor to replace the "pen-and paper" paradigm for the annotating needs. Different systems and tools of annotation are developed such as: iAnnotate (Plimmer et al,2010), u-Annotate (Chatti et al,2006), YAWAS⁶, iMarkup (2013), etc. Such tools enable readers to annotate their digital documents with free form annotations similarly to "pen-and paper" case. iAnnotate for example is an annotation tool for android system and it enables users to add annotations with the pencil, highlighter, and note tools.

Recently we intend talking about new products for reading such as the tablet. With the aid of such devices, the user may interact easily with a digital document and enter her annotations as he do in the case of "pen-and-paper". Thus, our findings is promising and original to be applied in such digital areas.

⁶Yawas is a free web annotation tool for Firefox and Chrome built on top of Google Bookmarks. Yawas enables you to highlight text on any webpage and save it in your Google Bookmarks account.

We believe it's the occasion to develop a system to attract those that have a curiosity to use annotation platforms and practices - from end users including scholars, scientists, journalists, public servants...etc. We expect our system enables readers to interact actively with their reading materials via annotation practices. Our goal is to use the traced annotations on the digital document to infer the annotator personality traits. Thus the expected system should contain free form annotation tool easy to be used, be able to infer user personality traits from captured annotation activity and refines the user traits profile by reference to new captured annotations. Based on the modelled profile we expect our system be able to offer services to users such as friendship recommendation based on similarity of users personality, customizing user interface using such predefined personas, sharing annotated documents...etc.

To achieve the personalization process we need to know certain user's features. Several research works prove that prediction of personality traits reveals a lot about a user's features. These findings was applied in several domains such as recommender systems (Nunes et al,2008) (Roshchina et al,2011) and the results is interesting.

The annotation as an unselfconscious practice constitutes a credible source of knowledge. In (Kalboussi et al,2013) the annotation activity is used to invoke the appropriate Web services to users. This proves that annotation is rich enough to be used differently.

In this paper we use annotation to infer such reader traits. That is a promising work and represents a new tendency to model user personality from human behaviour.

In other hand, let's be objective, there are some limitations to this work. The most important issue is the sample size as we expect more significant results with a larger sample. This limitation may be due to the dependency to "pen-and-paper" approach which prevents us to benefit from the population of readers over the web. In addition, people are not interested to participate in our experimentation unless they are motivated. We expect resolving these issues in future works.

Finally, our work is the first step to study the relation of reader annotation practices to human personality traits. So much perspectives can be subjects of future works such as studying factors which are likely to influence annotating behaviour such as familiarity with annotation tools and interest in the content topic. Also our research can be extended to study the possibility to predict human traits through social annotations. These avenues and others are very interesting

and represent an opened future directions which needs more investigations.

7 CONCLUSION

In this paper, we have shown that such users' personalities traits can be predicted from their annotation practices. With this ability of prediction many opportunities are opened which suggests future directions in variety of areas such as user modeling, recommender systems, user interface design and so on areas relative to personalization research domain. Furthermore, this work bridges the gap between the reading and the personality research domains and it remains an open research questions to see whether personality can also be predicted using other potentially features of reading activity as well as the influence of such environmental factors on human annotation behaviour.

REFERENCES

- Agosti, M. Ferro, N., 2003. *Annotations: Enriching a Digital Library*. In: Proceedings of the 7th European Conference (ECDL) Trondheim, Norway. Springer Pages 88-100.
- Bachrach, Y. Kosinski, M. Graepel, T. Kohli, P. Stillwell, D., 2012. *Personality and Patterns of Facebook Usage*. In: Web Science12 Proceedings of the 3rd Annual ACM Web Science Conference. ACM New York, NY, USA, Pages 24-32.
- Burger, J. M., 2011. *Personality*. Editor Wadsworth, USA.
- Barrick, M. R. Mount, M.K., 1991. *The big five personality dimensions and job performance: a meta-analysis*. Personnel psychology, Volume 44, Issue 1, Pages 126.
- Chatti, M. A. Sodhi, T. Specht, M. Klamma, R. Klemke, R., 2006. *u-Annotate: An Application for User Driven Freeform Digital Ink Annotation of E-Learning Content*. In: Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies (ICALT), Washington, DC, USA. Pages 1039-1043.
- Eswaran, S. Aminul Islam, Md. Dayang Hasliza, M.Y., 2011. *A Study of the Relationship between the Big Five Personality Dimensions and Job Involvement in a Foreign Based Financial Institution in Penang*. In: International Business Research. Vol. 4, No. 4.
- Johnson, J.A. *The IPIP-NEO: International Personality Item Pool Representation of the NEOPI-R*, viewed 8 June 2013, <http://www.personal.psu.edu/~j5j/IPIP/ipipneo300.htm>
- Golbeck, J. Robles, C. Edmondson, M. Turner, K., 2011. *Predicting Personality from Twitter*. In: Privacy, Security, Risk and Trust PASSAT, 2011 IEEE Third International Conference on Privacy, Security, Risk, and Trust, and 2011 IEEE Third International Conference on Social Computing.
- Kalboussi, A. Mazhoud, O. Hadj Kacem, A., 2013. *Annotative Activity as a Potential Source of Web Service Invocation*. In: Proceedings of the 9th International Conference on Web Information Systems and Technologies (WEBIST), Aachen, Germany. SciTePress Pages 288-292.
- Nunes, M.A. Cerri, S.A. Blanc, N., 2008. *Improving Recommendations by Using Personality Traits in User Profiles*. In: International Conferences on Knowledge Management and New Media Technology, Graz, Austria.
- Nunes, M.A. Cerri, S.A. Blanc, N., 2008. *Towards User Psychological Profile*. In: IHC '08 Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems. ACM Pages 196-203.
- Nunes, M.A., 2008. *Recommender Systems based on Personality Traits*. PhD Thesis, University of MONTPELLIER 2.
- Marshall, C., 2010. *Reading and Writing the Electronic Book*. Editor Gary Marchionini, University of North Carolina, Chapel Hill.
- Plimmer, B. Hsiao-Heng Chang, S. Doshi, M. Laycock, L. Seneviratne, N., 2010. *iAnnotate: Exploring Multi-User Ink Annotation in Web Browsers*. In: Proceedings of the 11th Australasian User Interface Conference (AUIC). ACM Pages 52-60.
- Peabody, D. De Raad, B., 2002. *The Substantive Nature of Psycholexical Personality Factors: A Comparison Across Languages*. Journal of Personality and Social Psychology, vol 83, issue 10; Pages 983-997.
- Roshchina, A. Cardiff, J. Rosso, P., 2011. *User Profile Construction in the TWIN Personality-based Recommender System*. In: Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP), IJCNLP. Pages 7379.
- Ryckman, R. M., 2008. *Theories of Personality*. Editor Thomson Wadsworth USA.
- Schmitt, D. Allik, J. McCrae, R. Benet-Martinez, V., 2007. *The geographic distribution of Big Five personality traits: Patterns and profiles of human self-description across 56 nations*. Journal of Cross-Cultural Psychology, vol 38, Pages 173-212.
- Tkalcic, M. Kunaver, M. Tasic, J. Košir, A., 2009. *Personality based user similarity measure for a collaborative recommender system*. In: Proceedings of the 5th Workshop on Emotion in Human-Computer Interaction-Real world challenges, Cambridge, UK. Pages 30-37.
- The iMarkup Annotation Tool: a Commercial tool, viewed 13 September 2013, <http://www.bplogix.com/support/imarkup-client.aspx>

Towards Automatic Building of Learning Pathways

Patrick Siehndel¹, Ricardo Kawase¹, Bernardo Pereira Nunes² and Eelco Herder¹

¹ *L3S Research Center, Leibniz University Hannover, Hannover, Germany*

² *Department of Informatics, PUC-Rio, Rio de Janeiro, RJ, Brazil*
{siehndel, kawase, herder}@L3S.de, bnunes@inf.puc-rio.br

Keywords: Learning Support, Learning Pathways, Digital Libraries.

Abstract: Learning material usually has a logical structure, with a beginning and an end, and lectures or sections that build upon one another. However, in informal Web-based learning this may not be the case. In this paper, we present a method for automatically calculating a tentative order in which objects should be learned based on the estimated complexity of their contents. Thus, the proposed method is based on a process that enriches textual objects with links to Wikipedia articles, which are used to calculate a complexity score for each object. We evaluated our method with two different datasets: Wikipedia articles and online learning courses. For Wikipedia data we achieved correlations between the ground truth and the predicted order of up to 0.57 while for subtopics inside the online learning courses we achieved correlations of 0.793.

1 INTRODUCTION

When learning about a new topic, especially in a domain that is new to the learner, it is not always directly clear in which order relevant resources can best be read or learned, ensuring that the basic concepts are introduced first, followed by more advanced material that elaborates on these concepts. This is commonly known as *Learning pathway*. In fact, a learning pathway is described as the chosen route, taken by a learner through a range of learning activities, which allows them to build knowledge progressively (Jih, 1996).

Our approach exploits latent concepts inside learning objects and, according to the estimated complexity of these concepts, provides a tentative ordering for a set of learning objects. The results provide learners with an ordered learning script to follow, similar to a course in which lectures are arranged in a specific order.

For our method, we exploit information from Wikipedia, which we use as an external knowledge base. Wikipedia contains over 4 million articles (concepts) that virtually cover all concepts that are relevant for referencing. Further, each Wikipedia article contains links to reference articles and it is manually categorized. We exploit a set of features extracted from Wikipedia and its category graph to estimate the complexity of a given text. Our methods are based on the assumptions that:

- Wikipedia categories contain a useful link structure for ordering objects based on their difficulty;
- Concepts that are mentioned inside Wikipedia articles provide useful background knowledge for understanding the meaning of an article.

Our method uses the Wikipedia Miner¹ toolkit for detecting concepts in the analyzed learning objects. The detected concepts are basically text snippets that can be related to a Wikipedia article. All Wikipedia articles belong to one or more categories, and these categories are organized in a graph structure. We use this graph structure for identifying categories that are more general and therefore supposedly known by a user.

The main aspect of our work is to help learners to identify a meaningful order of given learning material. An example: in mathematics, it is obvious that learning basic principles like summing or dividing should come before starting with topics such as ‘curve sketching’. Essentially, the problem we aim to solve can be summarized as follows: given a set of learning objects, we bring them into a reasonable order, to help learners finding a good starting point as well as a good way through the provided material.

The rest of the paper is organized as follows: In Section 2, we discuss related work on the topics of learning object recommendation and ordering. The proposed method is explained in detail in Section 3.

¹<http://wikipedia-miner.cms.waikato.ac.nz/>

The experimental evaluation of the whole process is presented in Section 4, where we used two different data sets to analyze the performance of our method: Wikipedia articles and online learning courses from Khan Academy. We conclude the paper in Section 5 by summarizing our main contributions.

2 RELATED WORK

A dynamic course generator is presented by Farrell et al. (Farrell et al., 2004). The course is assembled based on keyword queries and the metadata of learning objects contained in a given repository. The sequence relies on the relationships that are manually assigned to each learning object and its classification (e.g. introduction, methodology or conclusion). Hence, the objects are selected and reordered according to a user query and its classification. Chen (Chen, 2008) present an evolutionary approach that uses a genetic algorithm to generate a learning plan. The genetic approach is based on a pretest performed by students, where missed concepts help in creating new learning plans, according to concepts and levels of difficulty of the learning objects. Our approach follows the dynamic nature of these approaches, since we only need an input concept to determine a learning object sequence.

Ullrich and Melis (Ullrich and Melis, 2009) order learning objects according to the learning goal of each student. For this, they classify objects into different classes, such as *illustrate* or *discover*, where the course is assembled by a sequence of examples or in depth.

In the areas of Intelligent Tutoring Systems and Adaptive Hypermedia the adaptive sequencing is common technique (Brusilovsky and Millán, 2007). In scenarios where metadata for the given learning objects is available systems like PASER (Kontopoulos et al., 2008) allow the calculation of a learning path. In our scenario we address informal learning situations in which this metadata is not available.

Another perspective on the sequencing of learning objects is discussed by Kickmeier-Rust et al. (Kickmeier-Rust et al., 2011), where they use a combination of a storytelling model and competence structures to identify the learning state of a student in games. By identifying the state of the student, they propose a new sequence of learning objects, while keeping the story lines. Limongelli et al. (Limongelli et al., 2009) present a framework to create personalized courses. The sequencing of learning objects is generated taking into account the cognitive state of the student and her learning style. Sequences change

according to the results obtained by the students, in order to cover a concept not understood. Missed concepts are identified through exercises during the learning process. Instead of discovering the learning state of students, we focus on a general applicable approach. Our approach identifies which topics are necessary to understand a topic independent of the student. On the one hand, we do not provide personalized learning paths; on the other hand, we overcome the cold-start problem where there is no a priori information of the students.

Champaign and Cohen (Champaign and Cohen, 2010) introduce a work based on student development after consuming a given learning object. Each student is assessed and the most successful sequence of learning objects is selected and recommended to students with similar profiles. Similarly, Knauf et al. (Knauf et al., 2010) focus on similar profiles to recommend similar learning paths. However, the similarity between students is based on learning models that describe the abilities of each student. A path taken by a successful student is recommended to another one with similar characteristics. In contrast, the goal of our approach is to recommend learning objects following the learning goals of a student; as the student selected a topic to learn, the sequencing is determined by knowledge and concepts needed to understand a learning object.

3 METHOD

Our method for ordering learning objects and providing background links is divided into two main steps. The first step is the annotation of the content with links to relevant Wikipedia articles. This step is described in more detail in Section 3.1. The second step exploits detected topics and Wikipedia as a knowledge base to calculate the order of learning objects in a given set.

3.1 Annotation and Features

For annotating the content of the given learning objects, we used the Wikipedia-Miner Toolkit (Milne and Witten, 2008). The tool annotates a text (links terms to articles) in the same way a human would do it in Wikipedia. With this information, based on the detected topics inside a given learning object, we calculate a set of features that indicate the complexity and relevance of a topic. The features we use for ordering the given objects are:

1. *Number of inlinks*: the number of Wikipedia articles that link to the detected topics.

2. *Number of outlinks*: the number of links to other articles contained by detected topics.
3. *Text Length of Linked Articles*: the length of the detected articles.
4. *Average Word Length of Linked Articles*: the average length of the words in the detected articles.
5. *Average Word Length of Learning Object*: the average length of the words in the learning object.
6. *Distance to Root of Linked Articles*: the average distance to the root categories of the articles.
7. *TF/IDF Score of Words in Linked Articles*: the TF/IDF values of the words inside the detected articles.
8. *TF/IDF Score of Words in Learning Object*: the TF/IDF values of the words inside the learning object.

The first two features are chosen based on the assumption that the number of inlinks and outlinks are indicators of the generality of a Wikipedia article: if many articles link to one page, it indicates that this concept is a basic (popular) concept. As in (Kamps and Koolen, 2009), inlinks and outlinks are deemed to be good indicators of an article's relevance to a given topic.

We also assume that the text length and the average words length are good indicators about how complex a topic is. Another important feature is the average distance of the related categories to the root node of the category tree. This feature is based on the assumption that more complex topics inside Wikipedia are deeper down in the category graph and is comparable to the generality feature in (Milne and Witten, 2012). The TF/IDF feature represents the assumption that words that rarely appear inside our corpus are related to more complex topics. All of our features are represented in four ways: we use the minimum, maximum, mean and standard deviation of each of these features to represent one learning object, which gives us a 32-dimensional float vector representing one learning object.

3.2 Learning to Order Objects

Our ordering approach is based on machine learning algorithms. The given features are used to generate a model that calculates a score for every learning object. This score indicates the estimated complexity of the concepts within the learning object. In our experiments, we used four different machine learning algorithms to produce the models. Two of these algorithms create a tree structure based on the given training data. In addition, we used a regression model and

a Support Vector Machine for regression to calculate the order of the given objects.

Note that the score is based on a comparison of learning objects, and this only makes sense if the learning objects cover related topics from a single domain. For example, answering the question if one should learn a topic like 'European History' before learning 'Linear functions' is out of the scope of this paper. Due to the different nature of different learning domains, the quality of the generated order is higher when only a single domain is considered. Our approach can help users to decide which object in a given set might be useful to be learned first, assuming that the objects are related per se.

4 EXPERIMENTAL EVALUATION

In this section we evaluate the performance of the proposed method by analyzing the quality of the predicted order of different sets of learning objects. We performed our evaluations with two different datasets: Wikipedia articles from different domains and a large set of online learning courses from the Khan Academy². We chose these datasets, as they contain elements that can be used as a ground truth that indicates how complex a given element is. For the Wikipedia articles, we chose the distance from the root node as an indicator for complexity. For the online course dataset, we exploited its hierarchical structure, which also indicates an order in which the elements should be learned.

4.1 Ordering Evaluation with Wikipedia Data

In this section, we describe the outcomes of our experiments with Wikipedia data. We show that there are useful correlations between the depth of a concept in the Wikipedia tree and other features that we use to define the complexity of a topic. For Wikipedia articles there is no predefined order that defines which article one should read first. We decided to take the distance to the root node of an article as an indicator for the complexity of the given topic. Every Wikipedia article belongs to at least one category, and based on the conventions how articles are added to categories, the articles should be added to the most specific category. Due to this, articles that belong to lower level categories cover in most cases more specific topics.

²<https://www.khanacademy.org/>

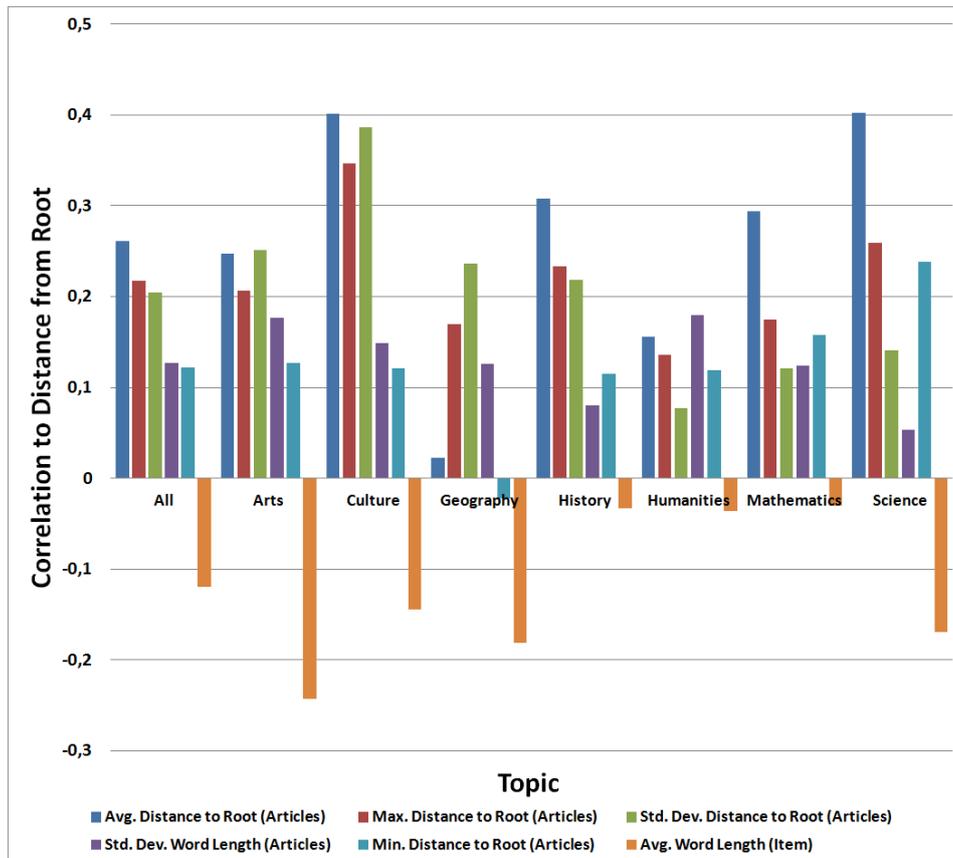


Figure 1: Correlation between features and distance to root of Wikipedia articles.

Table 1: Average distance to the root of articles from different categories.

| | | | | |
|----------------------|------------|-------------|-----------|---------|
| Category | Arts | Culture | Geography | History |
| Avg. Distance | 4.329 | 5.468 | 5.726 | 4.436 |
| Category | Humanities | Mathematics | Science | Average |
| Avg. Distance | 4.869 | 3.321 | 4.1 | 4.06 |

4.1.1 Dataset

In order to understand how the different features correlate and to get a first overview, we analyzed sets containing 500 articles from different Wikipedia Main Topics. The main topic is defined by following the category graph up to the first level of categories. The categories we chose to analyze are: ‘Arts’, ‘Culture’, ‘Geography’, ‘History’, ‘Humanities’, ‘Mathematics’ and ‘Science’. All of the mentioned categories also have a relation to topics taught in school and are therefore of special interest.

4.1.2 Ordering Wikipedia Articles

For learning an order inside the Wikipedia articles, we started by analyzing the distance to the root of articles belonging to different categories. As results show

in Table 1, different categories have different average distances to the root node. This is caused by the singular link and category structures inside the different categories. In comparison to ‘Mathematics’, which seems to have a relative flat category graph, we see that the average distance to the root for ‘Geography’ articles is much higher. Due to the large differences between the different categories, we decided to also analyze the correlations between the distance to the root and the calculated features for each category separately. The results for 6 of the features we analyzed is shown in Figure 1.

Overall we analyzed the correlations between 33 different features gathered from an article and its distance from the root of the category graph. Since our primary goal is to calculate an optimal order in which items should be learned, we analyzed how our fea-

Table 2: Results of predicted distance to root for Wikipedia articles using Machine Learning Algorithms.

| | SMOReg | M5P | Additive Regression | Bagging |
|--------------|---------------|---------------|---------------------|---------------|
| All Articles | 0.4878 | 0.5004 | 0.4422 | 0.5054 |
| Arts | 0.3587 | 0.4019 | 0.3611 | 0.3836 |
| Culture | 0.5253 | 0.509 | 0.5076 | 0.5213 |
| Geography | 0.0502 | 0.0027 | 0.3591 | 0.3835 |
| History | 0.2819 | 0.2516 | 0.267 | 0.2056 |
| Humanities | 0.0076 | 0.213 | 0.1777 | 0.2373 |
| Mathematics | 0.0907 | 0.4225 | 0.306 | 0.4313 |
| Science | 0.074 | 0.5704 | 0.5309 | 0.5478 |

tures correlate with the complexity of the Wikipedia articles. We divided the articles in two groups, based on their positions inside the category graph of Wikipedia. The first group consists of basic articles (distance<4), while the second group consists of advanced articles (distance≥4). Figure 1 shows the correlations between these groups and six features. The singularities between the different categories indicate that learning objects of each category may require different strategies.

It is noteworthy to mention that the feature “Distance to Root (Article)” is the most important feature. This feature is calculated based on the links to other articles inside the article that we want to rank. The positive correlations of maximum, minimum, and average show that articles that are already deep inside the Wikipedia category graph tend to have links to articles that are also deep inside this graph.

Another noteworthy fact is that the average word length inside the ranked articles has a negative correlation with the group index. This indicates that longer words (on average) tend to be in articles that are higher in the tree; this was not expected, as we expected to find longer words in articles that are deeper in the category tree.

The correlations found between the distance to the root of an article and the several features that we extracted from the articles indicate that it is possible to calculate an order for learning objects. To further analyze how well these features can be used to predict the complexity of a given text, we used machine learning algorithms to predict the actual distance to the root of a given article based on all the extracted features. The algorithms used are all integrated in Weka³ (Hall et al., 2009). We used SMOreg (Shevade et al., 2000), which is an implementation of a Support Vector Machine for regression, M5P (Quinlan, 1992) (Wang and Witten, 1997), which implements algorithms for creating M5 Model trees and rules, AdditiveRegression (Friedman and (y X)-values, 1999), which is an improved regression-based classifier, and a Bagging-

Algorithm (Breiman, 1996) based on a RepTree algorithm.

We predicted the distance for every category on its own and for all articles of the different categories together. The results are based on a 10-fold cross validation and shown in Table 2. We see that, for different categories, different algorithms produce the best result. On average, the Bagging-based approach produced the best results. Additionally, this algorithm shows a very low standard deviation over the different categories. In general, we see that the distance to the root for articles of the topics ‘Arts’, ‘Humanities’ and ‘Geography’ is harder to predict than in the cases of articles from ‘Science’, ‘Mathematics’ and ‘Culture’.

In summary, our first set of experiments shows that it is possible to predict a meaningful order for Wikipedia articles based on features extracted from Wikipedia’s link structure and the textual features within these articles.

4.2 Evaluation with Online Learning Data

In addition to the evaluation on Wikipedia Data, we performed an analysis of the proposed method on a real world dataset of learning courses. While the outcomes of the first experiments proved that the assumption that features gathered from text snippets and related Wikipedia articles can be used to calculate the complexity of given texts (by means of the distance of the article to the root node), we now use the given order of a set of learning objects as ground truth.

4.2.1 Dataset

The dataset used for this series of experiments was extracted from the online courses of Kahn Academy⁴. We analyzed the text of 2508 different lectures related to the main topics ‘Math’, ‘Science’ and ‘Humanities’. These items are organized in a three-level hierarchy: the first level is a general category like ‘Sci-

³<http://www.cs.waikato.ac.nz/ml/weka/>

⁴<https://www.khanacademy.org/>

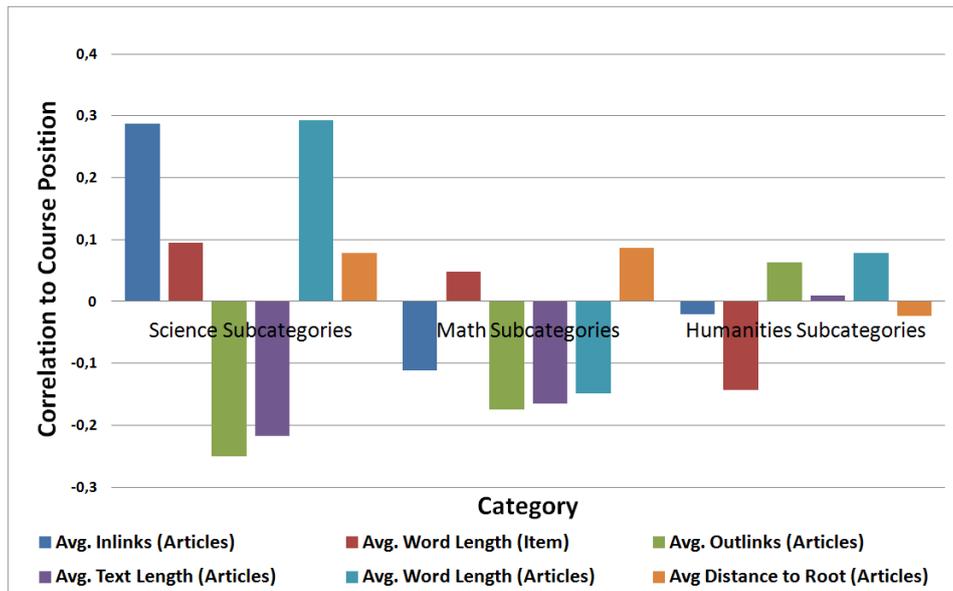


Figure 2: Correlation between features and learning object locations inside a course.

ence’; inside this category, there are different groups like ‘Chemistry’ or ‘Biology’. Below this level are the actual courses, like ‘Cell division’ or ‘Oxidation reduction’. The learning objects within a course are manually ordered in a meaningful way, representing the order in which a student is supposed to learn. Overall, we analyzed 110 different learning courses. Statistics on these courses are shown in Table 3

We chose to take the given order of the objects inside a course as ground truth for evaluating our approach. Calculating an order for a higher level does not make sense for all the given objects. For example, it is hard to say that ‘Biology’ should be learned before ‘Chemistry’, or that the ‘Industrial Revolution’ has to be learned before ‘Art History’, but when learning about matrices it seems to be useful to learn ‘Matrix multiplication’ before ‘Determinant calculation’.

4.2.2 Ordering Learning Objects

We started the analysis of learning objects in the same way as we did for Wikipedia articles: by analyzing correlations between the order of objects and the calculated features. The results showed us that the order of these items follows a more complex structure that is hard to grasp by just taking into account the linear relations between the order of the objects and the values of the calculated features.

Figure 2 displays the correlation values between learning objects and different features. We can see that with the shown features we do not obtain the same correlations for all kind of topics as we got for the Wikipedia articles. A closer look at the anno-

tated articles revealed that this is most likely caused by noise inside the transcripts of the online courses. This noise originates from the fact that the transcripts only represent the spoken content of the video lectures, which is hard to understand without the whole content of the video. Combined with a fair number of non-relevant remarks that were still included in the transcript, the quality of the extracted articles is not as good as in the previous experiment. Despite this drawback, for many of the features there are clear relations between the features and the location inside the course. We decided to perform the same tests as before to calculate the actual position of the learning objects inside the courses.

The results of this series of experiments are displayed in Table 4. The results were produced using all mentioned features using a 10-fold cross validation.

The highest overall achieved correlation between the actual position and the calculated position was at 0.554, when the algorithm is applied on all available learning objects. When training and testing on subcategories of the data, we achieve results of up to 0.793. The results differ strongly between the different domains of the online courses. For the elements of the domain ‘Humanities’ none of the tested algorithms achieved good results, while the order of ‘Science’-related elements was relatively well calculated by all algorithms. We also see that not all different algorithms are in the same way suitable for predicting the actual rank of the items. On average, the best results were achieved using the Bagging approach.

Table 3: Statistics on the Learning Object Dataset.

| Main Category | #Groups | #Courses | Avg. Items per Course |
|---------------|---------|----------|-----------------------|
| Humanities | 2 | 18 | 30.92 |
| Mathematics | 5 | 40 | 33.38 |
| Science | 4 | 47 | 10.76 |

Table 4: Results of predicted positions of learning objects using Machine Learning Algorithms.

| | SMOReg | M5P | Additive Regression | Bagging |
|-------------|--------|--------------|---------------------|--------------|
| All LOs | 0.292 | 0.338 | 0.408 | 0.554 |
| Mathematics | 0.094 | 0.365 | 0.397 | 0.416 |
| Science | 0.357 | 0.779 | 0.71 | 0.793 |
| Humanities | 0.056 | 0.141 | 0.135 | 0.127 |
| Wikipedia | 0.488 | 0.500 | 0.442 | 0.505 |

4.3 Discussion

The series of conducted experiments shows that the proposed method can be used for calculating the complexity of a given topic, based on text features and features extracted from Wikipedia. Additionally, there are evidences that for some categories it is harder to predict its complexity than for others. Especially content from the area of Humanities seems to be harder to order than content from disciplines like Mathematics or Science. This might be due to a more complex structure of the underlying content: in Mathematics, the order in which elements need to be learned is much clearer, due to the fact that concepts build up on one another. By contrast, in disciplines like History, this is in most cases not true.

5 CONCLUSION

In this paper, we presented a method for ordering learning objects based on the complexity of the covered content. The proposed method is based on features that are extracted from the original items, as well as from the knowledge stored in Wikipedia. By using Wikipedia, we exploit a knowledge base that is constantly updated and freely available. We analyzed the performance of the method on two different datasets, and achieved correlations between the ground truth and the predicted values of up to 0.793 for special topics of learning courses. The results show that text-based learning material can automatically be sorted in a meaningful order. However, the quality varies, depending on the domain and the textual quality of the elements. For example, written text from Wikipedia is easier to order than noisy video transcripts.

The results of the experiments also showed that the proposed method works better with domains like Mathematics or Science compared to domains like

Humanities or History. In general it seems to be useful to train different models for different domains since the values of some features vary over different domains.

The proposed order, as provided by our method, can help learners to find a good starting point for their learning pathways inside a set of learning resources. Also, it might help them to choose how to continue their learning process once a lesson has been learned or a resource has been visited. In addition, the methods may help teachers to analyze how the complexity of their courses evolves over time, which may help them to find a more suitable order for the elements they are teaching. A big advantage of the proposed method is that no metadata is required for calculating an order. This allows to incorporate every kind of textual resource into the learning process.

As future work we plan to build a model that can identify prerequisite knowledge for given learning courses. This will allow teachers and learners to better build a background knowledge for teaching/learning activities.

ACKNOWLEDGEMENT

This work has been partially supported by the European Commission under ARCOMEM (ICT 270239) and QualiMaster (ICT 619525).

REFERENCES

- Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.
- Brusilovsky, P. and Millán, E. (2007). User models for adaptive hypermedia and adaptive educational systems. In *The adaptive web*, pages 3–53. Springer-Verlag.

- Champaign, J. and Cohen, R. (2010). A model for content sequencing in intelligent tutoring systems based on the ecological approach and its validation through simulated students. In Guesgen, H. W. and Murray, R. C., editors, *FLAIRS Conference*. AAAI Press.
- Chen, C.-M. (2008). Intelligent web-based learning system with personalized learning path guidance. *Computers & Education*, 51(2):787 – 814.
- Farrell, R. G., Liburd, S. D., and Thomas, J. C. (2004). Dynamic assembly of learning objects. In *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, WWW Alt. '04, pages 162–169, New York, NY, USA. ACM.
- Friedman, J. H. and (y X)-values, O. K. (1999). Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 38:367–378.
- Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. (2009). The WEKA data mining software: an update. *Special Interest Group on Knowledge Discovery and Data Mining Explorer Newsletter*, 11(1):10–18.
- Jih, H. J. (1996). The impact of learners' pathways on learning performance in multimedia computer aided learning. *J. Netw. Comput. Appl.*, 19(4):367–380.
- Kamps, J. and Koolen, M. (2009). Is wikipedia link structure different? In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 232–241, New York, NY, USA. ACM.
- Kickmeier-Rust, M., Augustin, T., and Albert, D. (2011). Personalized storytelling for educational computer games. In Ma, M., Fradinho Oliveira, M., and Madeiras Pereira, J., editors, *Serious Games Development and Applications*, volume 6944 of *Lecture Notes in Computer Science*, pages 13–22. Springer Berlin Heidelberg.
- Knaufl, R., Sakurai, Y., Takada, K., and Tsuruta, S. (2010). Personalizing learning processes by data mining. In *Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conference on*, pages 488–492.
- Kontopoulos, E., Vrakas, D., Kokkoras, F., Bassiliades, N., and Vlahavas, I. (2008). An ontology-based planning system for e-course generation. *Expert Systems with Applications*, 35(1):398–406.
- Limongelli, C., Sciarrone, F., Temperini, M., and Vaste, G. (2009). Adaptive learning with the ls-plan system: A field evaluation. *Learning Technologies, IEEE Transactions on*, 2(3):203–215.
- Milne, D. and Witten, I. H. (2008). Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 509–518, New York, NY, USA. ACM.
- Milne, D. and Witten, I. H. (2012). An open-source toolkit for mining wikipedia. *Artificial Intelligence*.
- Quinlan, J. R. (1992). Learning with continuous classes. In *Proceedings of the 5th Australian joint Conference on Artificial Intelligence*, volume 92, pages 343–348. Singapore.
- Shevade, S. K., Keerthi, S. S., Bhattacharyya, C., and Murthy, K. R. K. (2000). Improvements to the SMO algorithm for SVM regression. *Neural Networks, IEEE Transactions on*, 11(5):1188–1193.
- Ullrich, C. and Melis, E. (2009). Pedagogically founded courseware generation based on htn-planning. *Expert Systems with Applications*, 36(5):9319 – 9332.
- Wang, Y. and Witten, I. H. (1997). Inducing model trees for continuous classes. In *Poster Papers of the 9th European Conference on Machine Learning (ECML 97)*, pages 128–137. Prague, Czech Republic.

A Survey on Challenges and Methods in News Recommendation

Özlem Özgöbek^{1,2}, Jon Atle Gulla¹ and R. Cenk Erdur²

¹*Department of Computer and Information Science, NTNU, Trondheim, Norway*

²*Department of Computer Engineering, Ege University, Izmir, Turkey*
{ozlemo, jag}@idi.ntnu.no, cenk.erdur@ege.edu.tr

Keywords: Recommender Systems, News Recommendation, Challenges.

Abstract: Recommender systems are built to provide the most proper item or information within the huge amount of data on the internet without the manual effort of the users. As a specific application domain, news recommender systems aim to give the most relevant news article recommendations to users according to their personal interests and preferences. News recommendation have specific challenges when compared to the other domains. From the technical point of view there are many different methods to build a recommender system. Thus, while general methods are used in news recommendation, researchers also need some new methods to make proper news recommendations. In this paper we present the different approaches to news recommender systems and the challenges of news recommendation.

1 INTRODUCTION

The increasing amount of data on the internet makes harder to find what we are really looking for. Even though the technologies like search engines and RSS readers help us, it is still hard to find the information we really want to get. On the other hand, we are not always sure about what we want to get. We can only search for what we know and we try to find some connections to the new information. But this approach of finding an item that the user will like mostly depends on the coincidences, the attention of the user to inspect the search results and it requires a lot of effort. Still there is a high possibility that the user could not finding the most suitable item for herself at the end.

Recommender systems are built to help us to easily find the most proper information on the internet. Unlike the search engines recommender systems bring the information to the user without any manual search effort. This is achieved by using the similarities between users and/or items. There are many methods to build a recommender system and these methods can be applied to many specific domains like shopping (e.g. Amazon), movies (e.g. Netflix) and music (e.g. Pandora Radio). Since each application domain has its own specific needs, the method used for recommendations differs.

As people are beginning to read news online more and more, it became a challenge to find the interesting news articles. Most of the users spend a lot of

time to find an interesting article on a single website or they just read the front page news which is not adequate. When we consider the different news sources on the internet, one can spend plenty of time just reading the news. News recommender systems aim to give the most relevant article recommendations to users according to their personal interests and preferences.

Recommending news articles is one of the most challenging recommendation tasks. The news domain differs from other domains in many ways. For example; the popularity and recency of news articles changes so fast over time. So focusing on the recency issue becomes more challenging than it is in other domains. Also some news articles may be connected with each other that the user may want to read the previous news items related to the one she already reads or she may want to keep informed about. Only learning user preferences can be an unsatisfactory solution to news recommendation. This is because the user may want to read a news article when she is not really interested in the subject but she thinks it is important. For example; wanting to read the news about elections even if she is not generally interested in politics. Also considering the high number of new articles published every hour increases the complexity of other challenges.

In this paper, we summarize the advances in this very special application domain of recommender systems which is news recommender systems. The paper is structured as follows: Section 2 gives an overview

of recommender systems. Section 3 summarizes all the challenges of recommender systems in news domain which includes some common challenges with general recommender systems. Section 4 discusses the different approaches of news recommender systems for particular challenges. Section 5, gives the discussions. And in section 6 the conclusions are provided.

2 RECOMMENDATION TECHNIQUES

There are different methods for recommending an item. Most commonly they are grouped into three categories: Content-based filtering, collaborative filtering and hybrid. It is possible to categorize the recommendation techniques differently. For example in (Burke, 2002) five categories are proposed for recommendation techniques. These are: Collaborative, content-based, demographic, utility-based and knowledge-based techniques. This categorization of recommendation methods is based on the background data included in the system, input data gathered from online user interaction and the algorithm used for the recommendation. Since the first categorization is more widely used we will discuss these three categories. There is also an alternative semantic approach (Cantador et al., 2008), in which semantic representations are added on top of other methods, we will not go into the details of this work in the rest of the paper.

2.1 Content-based Filtering

In content-based recommendations, the properties of items are used to make recommendations. Items which have similar properties with the user's previous preferences are recommended to the user. Thus, for this technique it is important to find the similarities between items. For example; to recommend a movie to the user, the content-based system should know about the user's past movie preferences and the similarity between movies. As discussed in (Adomavicius and Tuzhilin, 2005) this approach has its roots in information filtering and information retrieval. Sometimes information filtering is used in the same meaning as content-based filtering (Lee and Park, 2007).

2.2 Collaborative Filtering

In collaborative filtering, recommendations are done by using the other people's preferences which are similar to the user's past preferences. Collaborative

filtering method can be divided into three as memory-based, model-based and hybrid methods. In **memory-based** (also called the neighborhood-based, user-based, heuristic-based) collaborative filtering method, user ratings are used to compute similarities between users/items. By using the statistical techniques it is aimed to find a similar user to the targeted user. After the similarities found the recommendations can be done by using different algorithms. In the **model-based** (also called item-based) method, a model is created for each user by using data mining and machine learning algorithms. Probabilistic methods can be used for recommendation predictions. Methods like Bayesian network and clustering are included in this method (Sarwar et al., 2001). The **Hybrid** collaborative filtering method uses both model-based and memory-based methods.

2.3 Hybrid Approach

These approaches use both content-based and collaborative filtering. Generally the aim of these kinds of approaches is to come up with solutions to the problems which occur with the use of a single approach. There may be different combinations of using the two methods together. (Burke, 2002) groups the hybridization methods into seven and defines how different methods can be joined.

3 PARTICULAR CHALLENGES IN NEWS RECOMMENDATIONS

For many people building a recommender system can be perceived as an easy task at first sight. But finding the proper item to recommend can be a tedious task that requires access to information about the user, the items and the general context. Personal preferences and interests tend to vary on the basis of age, culture, gender and personality, and they also change over time. A successful recommender system needs to address a number of intrinsic challenges that each constitute a research field. In the news domain, because of the dynamic properties of news items some challenges have more importance than the others. The challenges we explain in this section are all related but not completely specific to news domain. Most of these challenges are the general challenges of recommender systems where some of the specific challenges (e.g. recency) may not be an issue in other domains.

- **Cold-start (First Rater, Ramp Up, Early**

- Rater) Problem:** The first-rater problem is one of the most common problems in recommender system collaborative filtering applications. Basically, it is the problem that the system cannot recommend new items if they do not have any clicks from other users. Or when there is no data about the completely new user then it is not possible to make recommendations for her.
- **Data Sparsity:** The matrices used for collaborative filtering can be very sparse when there are not enough ratings from users. The possibility of data sparsity increases if the number of columns or rows is much higher than the other. For example; if the number of items is much more than the number of users then it requires too many ratings to fill the item-user matrix. Data sparsity causes a decrease in the performance of the system.
 - **Recency:** Recency is one the most important challenges in news recommendation domain. Most of the users want to read fresh news instead of old dated articles. So the importance of news items decreases in time. On the other hand, some news articles may be connected with each other that the user may want to read the previous news items related to the one she already reads or he/she may want to keep informed about that subject (Li et al., 2011).
 - **Implicit User Feedback:** User feedbacks are quite important to make more precise recommendations. Without explicit feedbacks it may not be possible to understand if the user liked the article she read or not (Fortuna et al., 2010). But it is not practical for the system to interact with the user continuously. So the system should be able to collect implicit feedbacks effectively while protecting the user privacy.
 - **Changing Interests of Users:** Another key challenge is predicting the future interests of users for better recommendations because people may have changing interests (Liu et al., 2010). For some domains like movie or book recommendations, the change of user interest happens more slowly. But for the news domain it is really hard to predict the changes. Also some people may read the news not because he/she interested in the topic in general but because she found it important.
 - **Scalability:** Recommender systems are aimed to serve many users, sometimes millions of users (Das et al., 2007) at a time. Also the number of items to be recommended can be very high. To build a really useful recommender system it is needed the system to be fast. In different news sources on the internet it is possible to find tens of new headlines within an hour. So in this dynamic environment of news, the news recommender system should have a fast and real time processing capabilities (Li et al., 2011). Independent from which approach is used, scalability is one the most important problems of recommender systems.
 - **Unstructured Content:** For the systems which require content information, it is hard to analyze the content, especially for the news domain. For better news recommendations, news items should be structured and machine readable (Saranya.K.G and Sadhasivam, 2012).
 - **User Modeling/profiling (Knowledge of User Preferences):** User profiling is an important component of recommender systems. To make more individual specific recommendations it is needed to construct a user profile. As it is stated in (Liu et al., 2010), (Das et al., 2007), (Saranya.K.G and Sadhasivam, 2012) there are many different approaches for user profiling.
 - **Gray Sheep Problem:** Since collaborative filtering recommends items according to the user's common interests with other users, it is not possible to recommend proper items to people whose preferences do not consistently agree or disagree with any group of people (Su and Khoshgoftaar, 2009). When the total number of users increases, the possibility of this problem occurring decreases (Borges and Lorena, 2010).
 - **Serendipity (Over-specialization, Portfolio) Problem:** This is the problem when the system recommends similar or the same items with the already recommended ones. For the news domain, a news item written differently in different news sources may be recommended by the recommender system as different articles. It is obvious that the users would not be happy to get the same or similar recommendations. The system should always be able to discover new items to recommend by avoiding the same items. In (Jaquinta et al., 2008) the problem is discussed in detail for content-based systems but it is also a problem for collaborative filtering systems (Borges and Lorena, 2010).
 - **Privacy Problems:** To make proper recommendations to a user, the system should know about the users' past preferences, interests and even the relations with other people. This requires the storage of detailed data about the user and the analysis of this data that can cause privacy issues (Garcin et al., 2013).
 - **Neighbor Transitivity:** Neighbor transitivity occurs when the database is very sparse. Even if

there are two users who have similar interests, the system cannot detect them because of the lack of ratings they have on similar items (Su and Khoshgoftaar, 2009).

- **Synonymy:** Same items can be named differently by separate resources and it is not possible for machines to understand that they refer the same item. For example; even if the “children’s movie” and “children’s film” have the same meaning, they can be treated as different items by the recommender system (Su and Khoshgoftaar, 2009).

4 APPROACHES OF NEWS RECOMMENDER SYSTEMS

In this section we explain the different approaches to the most addressed challenges of news recommender systems. Some of these challenges are not completely specific to the news domain. But in most of the previous works about news recommender systems these challenges are prioritized and mostly addressed. The summary of which approach addresses to solve which particular challenge can be seen in Table 1. The term N/A is used for defining that particular challenge is not addressed or it is unknown if the challenge is addressed or not in that work.

4.1 Cold-Start Problem

Since collaborative filtering finds similarities between different users’ and makes recommendations by using the different preferences of similar users, it is impossible to recommend a new item which is not evaluated yet. Another aspect of this problem is that it is impossible to recommend any items for completely new users. Cold-start problem is the most common problem for the applications that use collaborative filtering. As it is mentioned in (Liu et al., 2010) for some researchers it is one the most important disadvantages of collaborative filtering approach. To solve this problem (Liu et al., 2010), (Fortuna et al., 2010) and (Lin et al., 2012) are proposed a hybrid method using the collaborative filtering and content-based filtering together. In (Liu et al., 2010), it is proposed to use the personalization for recommending new articles which is building a profile for the user’s genuine interests. In (Lin et al., 2012), to handle the cold-start problem, the system includes the opinions of chosen experts (who uses the social networks have significant influence on new users). So that the system can make recommendations to a new user. By using the TF-IDF method new items can also be recommended. In (Fortuna et al., 2010), it is proposed another approach that

is grouping the users as an old or a new user according to the number of articles they read. For each group of users, a separate model is trained for predicting the most interesting news category. And the top new article from each filtered category is recommended to the user. (Tavakolifard et al., 2013) proposes an architecture which considers the users’ long term preferences, short term preferences and the current context. So that cold start problem can be eliminated by using the current contextual information for first recommendations. In (Lee and Park, 2007) it is checked that if the user is new or not. If it is a new user than she is temporarily placed in a similar segment on the basis of demographics and the first recommendations done according to the preferences of that demographic segment. (Garcin et al., 2013) proposes a system which works for anonymous users. When a user starts to read a news item the system generates recommendations and during the session of the user the system updates the model and makes better recommendations.

4.2 Recency

We see that nearly in all the works done on news recommendation the importance of recency is addressed. In (Yeung and Yang, 2010) recency mentioned as an important property of a recommender system and it is proposed a proactive news recommender system for mobile devices. Since the environment for a mobile user is highly dynamic it is a challenge to deliver the most proper and recent information to the user. In the proposed system a Hybrid P2P system is used to deliver the just-in-time information to users’ mobile device. A pure P2P system is not suitable for mobile devices since it requires lots of communication with other devices. In the proposed architecture, mobile device connects one of the servers in the network and sends the context information. Recommendation is done by the server (which gets the recent news articles constantly from RSS) by using this context information of the user which includes user profile, location, usage patterns, peer ratings etc. As a different approach, (Wen et al., 2012) includes the time factor in recommendation process. In this approach, to recommend a news item, time factor is taken into account as a coefficient in addition to the user interest and preference models. Similarly, in (Lee and Park, 2007) weights of articles is calculated by the degree of importance and recency of that article. In (Fortuna et al., 2010), news categories are determined for each user according to the user’s preferences and the newest article of each category is selected for recommendation. On the other hand, in (Li et al., 2011) it is constructed a news profile for news items which includes dynamic

Table 1: Different works on news recommendation with the challenges they addresses to solve.

| | Cold-start | Recency | Implicit Feedback | Changing User Interest | Scalability | Data Sparsity |
|------------------------------------|------------|---------|-------------------|------------------------|-------------|---------------|
| (Yeung and Yang, 2010) | N/A | ✓ | ✓ | ✓ | N/A | ✓ |
| (Wen et al., 2012) | N/A | ✓ | ✓ | ✓ | N/A | N/A |
| (Liu et al., 2010) | ✓ | ✓ | ✓ | ✓ | N/A | N/A |
| (Resnick et al., 1994) | N/A | ✓ | N/A | N/A | ✓ | N/A |
| (Lee and Park, 2007) | ✓ | ✓ | ✓ | ✓ | N/A | N/A |
| (Li et al., 2011) | ✓ | ✓ | ✓ | N/A | ✓ | N/A |
| (Das et al., 2007) | N/A | ✓ | ✓ | ✓ | ✓ | N/A |
| (Fortuna et al., 2010) | ✓ | ✓ | ✓ | N/A | N/A | N/A |
| (Tavakolifard et al., 2013) | ✓ | ✓ | ✓ | ✓ | N/A | N/A |
| (Saranya.K.G and Sadhasivam, 2012) | N/A | ✓ | ✓ | ✓ | ✓ | ✓ |
| (Garcin et al., 2013) | ✓ | ✓ | ✓ | ✓ | ✓ | N/A |
| (Lin et al., 2012) | ✓ | N/A | N/A | N/A | N/A | ✓ |

characteristics like recency and popularity. (Liu et al., 2010) considers the news trends to make proper recommendations. Since news trends mostly composed of recent news it can also be taken as a challenge of recency.

4.3 Implicit Feedback

To predict the future interests of a user and to make proper recommendations, the system needs to know the past interests of the user. There are two ways of learning about the past interests of a user: Explicit and implicit feedbacks. To collect explicit feedbacks it is needed to interact with the user continuously and ask if the user liked the item or not, how much she liked it and maybe other questions about the system in general. Both for the users and for the system it is not practical to continuously interact with the user. Especially in mobile devices it is hard to manually collect personal information (especially textual information) from the user. (Yeung and Yang, 2010) So it is desired to make user profiling and filtering automatically. The system should be able to collect implicit feedbacks effectively while protecting the user privacy. In (Garcin et al., 2013), it is stated that to overcome the lack of data about the users most systems require the users to have logged in the system which can cause privacy issues. Implicit feedbacks are mostly taken from the log analysis of users' history. In (Liu et al., 2010), to predict the future user interests, a large-scale log analysis is done over the registered users' history data and the change of user interests are observed. Similarly in (Wen et al., 2012) user's interest and preference models are constructed by using the user's navigational data. Also in (Tavakolifard et al., 2013) the user's behaviors are used for detecting some preferences of the user. In (Lee and Park, 2007) regular analysis is done over the history of the user and the system learns

about various preferences.

4.4 Changing User Interest

It is known that as time passes the interests of people change. The preferences of people about movies, music or books generally show a slight difference within short periods of time. But in the news domain it is again very different from other domains. The news reading preferences of people can be affected by on going circumstances in the world as well as their age, cultural level and even their mood. So, predicting future interests of users can be a real challenge for news recommendation. (Liu et al., 2010) addresses this challenge and proposes an architecture to predict the future user interests. To do this a hybrid news recommender system is proposed where a large-scale log analysis is done over the registered users' past activities. Click distribution over different news categories is computed both for individuals and groups in a monthly basis and the change of user interests are observed. For prediction of user interests, Bayesian framework is used. Then the predictions are used in information filtering method. To make proper recommendations it is combined with the existing collaborative filtering method. In (Lee and Park, 2007), the change of user interests are tracked by observing and comparing the long-term and short-term preferences. The comparison of coefficients of long-term and short-term preferences changes the weight of category preferences. So if there is a change in the category preference of a user over time, it can be used for making proper recommendations. (Wen et al., 2012) creates a model for the user's degree of interest to a specific topic by analyzing the navigational data (frequently visiting a web page related with a specific topic shows the user is interested in that subject) and then it is updated as the user keeps browsing

web pages. To keep track of the changing user interest (Saranya.K.G and Sadhasivam, 2012) proposes two kinds of user profiles: Static and dynamic user profiles. Static user profile includes the user's sign up information like user name and favorite topic where the dynamic user profile is constructed by using the implicit user data in every session.

4.5 Scalability

Since scalability problem applies to every computer related system it is also one of the most important challenges in recommender systems. If we want to build a useful recommender system it is obvious that it must be scalable. In news domain, scalability problem combines with other challenges which makes the news recommenders more challenging to build. In (Li et al., 2011) it is aimed a scalable news recommendation system by clustering the news articles and eliminating the unnecessary similarity computations. So that the system spends less time for computation and can response faster. In (Das et al., 2007) to be able to serve millions of users they proposed a new MinHash (a probabilistic clustering method) based user clustering algorithm, redesigned the PLSI (Probabilistic Latent Semantic Indexing, a model for performing collaborative filtering) as a MapReduce (a model for computing large scale data on clusters) computation and used item covisitation technique (a method for determining the user-item relations) for a more scalable system. (Saranya.K.G and Sadhasivam, 2012) discusses the need of scalability in efficient recommender systems. And the Hadoop framework handles the issues like reliability and scalability in their applications. Another different approach for news recommendation includes context trees. In (Garcin et al., 2013), it is discussed the scalability problem does not occur within this approach since it requires only one tree and the tree structure is very limited because of the applied context constraints.

4.6 Data Sparsity

Even though it is one of the most important challenges of collaborative filtering method, data sparsity is not addressed in most of the news recommender system approaches. Data sparsity occurs when the number of users or items are much higher than the other one. In this case when the user-item matrix is constructed, the matrix would be very sparse. In (Yeung and Yang, 2010), it is discussed that using Bayesian Network makes it easy to eliminate the data sparsity. On the other hand in (Saranya.K.G and Sadhasivam, 2012) HBASE (a non-relational distributed

database) is used to store the data where it can provide fault tolerant storage for sparse data. Another solution for data sparsity is to use the hybrid approach (Lin et al., 2012).

5 DISCUSSION

News recommendation is a specific domain in recommender systems which has special challenges and characteristics. Even though some of the challenges are shared with recommender systems in general, others may require different approaches to solve. As it is seen in Table 1, different works address to solve particular challenges. We see that all the approaches try to come up with solutions to as much challenges as possible. Some of them highlight one or two challenges and address the others as secondary challenges.

Since they are addressed and solved nearly in all of the works, recency and implicit feedback challenges seem the easiest ones to solve. For recency, most approaches prefer to recommend the latest headlines. Calculating the recency by using a coefficient which decreases in time is also another commonly used solution (Wen et al., 2012), (Lee and Park, 2007). Implicit feedback is also one of the most addressed challenges. It highly depends on the data extraction from users' navigational history or log analysis. The analysis and storage of this huge amount of data about users can cause privacy issues when it is required the users to log in to the system (Garcin et al., 2013). Also it reduces the scalability which is another important challenge. As we can see in Table 1, nearly all approaches have solutions for these two challenges. On the other hand, we see that scalability and data sparsity are the challenges which less solutions are offered. For some approaches like (Tavakolifard et al., 2013) data sparsity challenge is not available since it is a problem only for collaborative filtering method. It is also possible to eliminate the problems which belong to only one of methods by using hybrid systems.

As we can see in Table 2, some of the methods have minor differences but even though there are similarities between methods some approaches like context tree approach (Garcin et al., 2013) really differ from others. Some approaches include only one of the filtering methods where others include both of them (hybrid methods). In approaches that use hybrid method, it is possible to see that they use or propose different algorithms. (Yeung and Yang, 2010) proposes a new method for news ranking which is called AHP (Analytic Hierarchy Process) model. AHP of-

Table 2: Methods used in different news recommender systems.

| | Type | Algorithm | Log Analysis |
|------------------------------------|-------------------------|---|--------------|
| (Yeung and Yang, 2010) | Hybrid | Bayesian Network, AHP | ✓ |
| (Wen et al., 2012) | Hybrid | TF-IDF, Naive Bayes Model | ✓ |
| (Liu et al., 2010) | Hybrid | Bayesian framework | ✓ |
| (Resnick et al., 1994) | Collaborative filtering | Matrix Correlation | - |
| (Lee and Park, 2007) | Collaborative filtering | Specific equations used for calculations | ✓ |
| (Li et al., 2011) | Hybrid | LSH (Locality Sensitive Hashing) MinHash, NLP, LDA (Latent Dirichlet Allocation) | ✓ |
| (Das et al., 2007) | Collaborative filtering | For model-based - Item covisitation, for memory-based - PLSI and MinHash | ✓ |
| (Fortuna et al., 2010) | Hybrid | For model-based - SVM (Support Vector Machine) | ✓ |
| (Tavakolifard et al., 2013) | Content-based filtering | TF-IDF, NLP, NER (Named Entity Recognition) | ✓ |
| (Saranya.K.G and Sadhasivam, 2012) | Hybrid | Adaptive user profiling, dynamic neighborhood calculation, document ranking calculation | ✓ |
| (Garcin et al., 2013) | Hybrid | Context tree, BVMM, LDA (Latent Dirichlet Allocation) | ✓ |
| (Lin et al., 2012) | Hybrid | TF-IDF, probabilistic matrix factorization models | - |

fers a solution to assign weights for different ranking factors. (Fortuna et al., 2010) proposes an SVM (Support Vector Machine) (a machine learning model) based news recommender system. (Saranya.K.G and Sadhasivam, 2012) proposes calculation methods for adaptive user profiling, dynamic neighborhood calculation and document ranking calculation.

Using context trees for news recommendation is proposed in (Garcin et al., 2013) where it is used together with LDA (Latent Dirichlet Allocation) and BVMM (Bayesian Variable-order Markov Model) algorithms and defined as a scalable and effective solution to most of the challenges. LDA is also used in (Li et al., 2011) for the representation of topic distributions in a text. Bayesian Network is the most widely used technique to model user interests (Yeung and Yang, 2010), (Wen et al., 2012), (Liu et al., 2010). Another technique used for user profiling is NLP (Natural Language Processing) (Li et al., 2011), (Tavakolifard et al., 2013). There are different tools used for NLP technique like GATE and Apache OpenNLP library. For news items clustering LSH (Locality Sensitive Hashing) and MinHash (a probabilistic clustering method, a scheme for LSH (Das et al., 2007)) techniques are used in (Li et al., 2011). MinHash is used together with PLSI (Probabilistic Latent Semantic Indexing) in (Das et al., 2007). In content based filtering TF-IDF (Term Frequency-Inverse Document Frequency) is one of the mostly used techniques. In addition to TF-IDF, (Tavakolifard et al., 2013) uses NER (Named Entity Recognition) model to identify the names and location.

In Table 2 we can see that nearly in all of the works it is done log (or click) analysis over the usage data. Most systems require to users log in to the system. They gather data both from the sign up process and from the actions of the user while she is using the system. The approach in (Garcin et al., 2013) does not require any log in to the system, thus it makes recommendations only for the active session without determining who the user is. The other systems learn about users and they make recommendations based on the long term preferences of the user. By using log analysis it is also possible to track the change of user interests.

As the number of researches grow in recommender systems and specifically in news recommendation domain, we see that the number of hybrid systems increases. Recent evaluations show that hybrid systems tend to outperform other systems.

Evaluation and quantitative comparison of different recommender systems are other challenging aspects of recommender system research. Even though there are several evaluation methods for measuring the performance of the system, it is hard to measure the qualitative aspects like user satisfaction. There are some reasons which makes the quantitative comparison of the references we addressed in this paper not possible for us. First, they use different evaluation metrics which are not comparable. Second, because of the challenges they solve are different, the systems are not identical to each other to make quantitative comparisons.

6 CONCLUSIONS

Beginning from the first half of 90's, recommender system research continues to grow in different application domains. Nowadays it is not hard to see a recommender system working in the background of the web site you have visited and recommends you music or shopping items. Even though these useful applications of recommender systems exist, there are still many challenges for a true personalized recommender system. As the number of online news sources increases it becomes harder for an end user to find what she is looking for. Thus the need for a news recommender system increases.

In this paper, the challenges and different methods of news recommendation domain are presented. It is pointed out which approaches solve different challenges and how they do this. Our current framework for comparing recommender systems in news domain deals with content-based, collaborative and hybrid recommendation approaches, though we intend to expand it with recent results from semantically based recommender systems. Including semantic representations in the recommendation process helps us to understand user needs and news content and can be valuable when several of the challenges above are addressed.

REFERENCES

- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749.
- Borges, H. L. and Lorena, A. C. (2010). A survey on recommender systems for news data. In *Smart Information and Knowledge Management*, pages 129–151. Springer.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370.
- Cantador, I., Bellogín, A., and Castells, P. (2008). News@hand: A semantic web approach to recommending news. In *Proceedings of the 5th international conference on Adaptive Hypermedia and Adaptive Web-Based Systems*, AH '08, pages 279–283. Berlin, Heidelberg. Springer-Verlag.
- Das, A., Datar, M., Garg, A., and Rajaram, S. (2007). Google news personalization: Scalable online collaborative filtering. In *WWW '07 Proceedings of the 16th international conference on World Wide Web*, pages 271–280.
- Fortuna, B., Fortuna, C., and Mladenic, D. (2010). Real-time news recommender system. In *ECML PKDD'10 Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part III*, pages 583–586.
- Garcin, F., Dimitrakakis, C., and Faltings, B. (2013). Personalized news recommendation with context trees. *CoRR*, abs/1303.0665.
- Iaquinta, L., Gemmis, M. D., Lops, P., Semeraro, G., Filanino, M., and Molino, P. (2008). Introducing serendipity in a content-based recommender system. In *Hybrid Intelligent Systems, 2008. HIS'08. Eighth International Conference on*, pages 168–173. IEEE.
- Lee, H. and Park, S. J. (2007). Moners: A news recommender for the mobile web. *Expert Systems with Applications*, 32(1):143–150.
- Li, L., Wang, D., Li, T., Knox, D., and Padmanabhan, B. (2011). Scene : A scalable two-stage personalized news recommendation system. In *SIGIR '11 Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 125–134.
- Lin, C., Xie, R., Li, L., Huang, Z., and Li, T. (2012). Premise: personalized news recommendation via implicit social experts. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1607–1611. ACM.
- Liu, J., Dolan, P., and Pedersen, E. R. (2010). Personalized news recommendation based on click behavior. In *IUI '10 Proceedings of the 15th international conference on Intelligent user interfaces*, pages 31–40.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM.
- Saranya.K.G and Sadhasivam, G. (2012). A personalized online news recommendation system. *International Journal of Computer Applications*, 57(18):6–14.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pages 285–295. ACM.
- Su, X. and Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4.
- Tavakolifard, M., Gulla, J. A., Almeroth, K. C., Ingvaldesen, J. E., Nygreen, G., and Berg, E. (2013). Tailored news in the palm of your hand: a multi-perspective transparent approach to news recommendation. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 305–308. International World Wide Web Conferences Steering Committee.
- Wen, H., Fang, L., and Guan, L. (2012). A hybrid approach for personalized recommendation of news on the web. *Expert Systems with Applications*, 39(5):5806–5814.
- Yeung, K. F. and Yang, Y. (2010). A proactive personalized mobile news recommendation system. In *Developments in E-systems Engineering*, pages 207–212. IEEE.

Combining Learning-to-Rank with Clustering

Efstathios Lempesis and Christos Makris

¹*Department of Computer Engineering and Informatics, University of Patras, Patras, Greece
{lebasis, makri}@ceid.upatras.gr*

Keywords: Ranking, Learning-to-Rank, Clustering, Relational Ranking, Web Information Filtering and Retrieval, Searching and Browsing, Text Mining.

Abstract: This paper aims to combine learning-to-rank methods with an existing clustering underlying the entities to be ranked. In recent years, learning-to-rank has attracted the interest of many researchers and a large number of algorithmic approaches and methods have been published. Existing learning-to-rank methods have as goal to automatically construct a ranking model from training data. Usually, all these methods don't take into consideration the data's structure. Although there is a novel task named "Relational Ranking" which tries to make allowances for the inter-relationship between documents, it has restrictions and it is difficult to be applied in a lot of real applications. To address this problem, we create a per query clustering using state of the art algorithms from our training data. Then, we experimentally verify the effect of clustering on them.

1 INTRODUCTION

Nowadays, due to the evolution of the web it is common knowledge that it is difficult to find the desired information, so it is important to have search engines intelligent enough to meet our demands. As the user issues queries, we deem the ranking problem for information retrieval as the demand to order the stored set of documents by relevance to these queries. Ranking appears in many information retrieval problems, such as web search retrieval, collaborative filtering, entity ranking, sentiment analysis and text summarization. There are two types of ranking problems: ranking creation and ranking aggregation (Li, 2011). Ranking creation exploits the content of the document (as it appears as a set of features) in order to create a ranked list of documents, while ranking aggregation fuses multiple ranking lists, in order to create a unified ranked list.

The ranking module is responsible for matching between queries and indexed documents. A well-defined ranking module processes incoming queries and provides a matching score between them and the stored documents. Due to the fast development of the web and the flood of information, it is also as important as ever to have efficient and effective rankers that can rank this glut of information according to the users' queries.

In recent years (Liu, 2011; Li, 2011) it has become possible to embrace machine learning

technologies in order to build effective rankers, exploiting the large number of available training data. This embracement initiated a new research area called learning to rank, that combines traditional rankers with machine learning techniques; this area has become one of the most active in the area of web information retrieval.

Learning to rank or machine-learned ranking (MLR) automatically constructs ranking models from training data in terms of a loss function; it can be phrased in different types of supervised or semi-supervised machine learning problems. The ranking model has as purpose to produce a proper ranked list in new queries by exploiting the training data lists of items with each list providing some partial order between its items. To grant this order either numerical scores, ordinal scores or binary judgments (degree of relevance) are provided. Its methods can be categorized as: the pointwise approach, the pairwise approach, and the listwise approach (Liu, 2011). These approaches differ according to the loss functions they employ. Regarding the pointwise approach, which can be considered as a classification or regression problem by learning the rank values of the documents, the input space consists of a feature vector for each discrete document and the output space consists of the relevance grades. The input space of the pairwise approach, which treats the pair of documents as independent quantities and learns a classification or

regression model to correctly order these pairs, consists of feature vectors of pairs of documents and the output space consists of the pairwise preference $\{+1, -1\}$ between each pair of documents. The input space of the listwise approach consists of a corpus of documents related to a single query and considers them as a training example. Its output space contains the ranked list of the documents. The main problem with the pointwise and pairwise approaches is that their loss functions are associated with particular documents while most evaluation metrics of information retrieval compute the ranking quality for individual queries and not for documents. The goal of the listwise approach is to maximize the evaluation metrics such as NDCG and MAP.

A lot of the real ranking procedures actually think of the relationship between the documents, but all of the proposed learning-to-rank algorithms, which belong to any of the above approaches, do not take this into account. We could imagine this connection as the relationships between the clusters, the parent-child hierarchy etc.

Similar to the toy example in Kurland's PhD thesis (Kurland, 2006), let $q = \{\text{computer, printer}\}$ be a query, and consider the documents:

d1 = computer, company, employ, salary
 d2 = computer, investment, employer, company
 d3 = taxes, printer, salary, company, employer
 d4 = computer, printer, disk, tape, hardware
 d5 = disk, tape, hardware, laptop
 d6 = disk, tape, floppy, cd rom, hardware

Both the documents and the query are represented using a vector space representation (Baeza-Yates and Ribeiro-Neto, 2011) and the weight for each term in a vector is its frequency within the corresponding document (or query). If we rank the documents regarding q , we may get the following ranking:

Ranked list = d4, d1, d2, d3, d5, d6 (d4 is the top retrieved document.)

However, since it is more rational to suppose that the fundamental topic of the query is "computer hardware" rather than "business", we would like to have d5 and d6 ranked as high as possible in the list. Clustering the documents using the scheme, where each document belongs to exactly one cluster, into two clusters, could result in the following clusters: $A = \{d1, d2, d3\}$, $B = \{d4, d5, d6\}$. If we took this clustering into account and applied the cluster hypothesis then d5 and d6 would be ranked higher than d1, d2 and d3. That is the desirable outcome, since d5 and d6, though not containing any of the terms that occur in q are more close to the query's

topic(computer hardware), than d1, d2 and d3, which contain one query term, but do not seem to discuss the query topic.

As another sign of the significance of clustering in (Zeng et al., 2004) it has been mentioned that existing search engines such as Google (www.google.com), Yahoo (<http://search.yahoo.com/>) and Bing (www.bing.com) often return a long list of search results, ranked by their relevancies to the given query. As a consequence, Web users must sequentially seek the list and examine the titles and snippets to discern their desired results. Undoubtedly, this is a time consuming procedure when multiple sub-topics of the given query are mingled together. They propose that a possible solution to this problem is to (online) cluster search results into different groups, and to enable users to recognize their required group.

Carrot2 (<http://search.carrot2.org/stable/search>) is a real illustration of this approach.

The aim of present work is to investigate whether it is possible or not to integrate into the learning-to-rank algorithm's procedure, without user intervention, the information that we gain by clustering following the well known *cluster hypothesis* of the information retrieval area (Kurland, 2006; Gan, Ma and Wu, 2007; van Rijsbergen 1984) and examine the results of this venture. Hence, after the off-line building of the clusters and during the algorithm's function we provide to each document the bonus that corresponds to its cluster. Through this procedure we build on the assumption that a document, which belongs to one cluster, will be near the other documents of its cluster at the ranked list. In a narrow sense, we estimate that the documents, which belong to the best cluster, will be at the top of the ranked list and as a consequence we will have better ranked lists and better measure metrics.

Before concluding the introduction we describe some basic notions:

The *BM25* weighting scheme (Robertson et al., 2004) is a ranking function used by search engines to rank matching documents according to their relevance to a given search query.

Mean Average Precision (MAP) (Baeza-Yates and Ribeiro-Neto, 2011) for a set of queries q_1, \dots, q_s is the mean of the average precision scores for each query.

DCG (Baeza-Yates and Ribeiro-Neto, 2011) measures the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated from the top to the bottom of the result list with each result's gain being discounted at lower

positions.

Precision (Baeza-Yates and Ribeiro-Neto, 2011) is defined as the fraction of the retrieved documents that are relevant. These values are typically evaluated at a given cut-off rank, considering only the topmost results; in this case it is called precision at k or $P@k$.

Finally, the paper is organized as follows. The algorithms under examination are presented in Section 2. In Section 3, we present our ideas and how we implemented them, while in Section 4 we present the clusters' creation and our key findings. In Section 5 we conclude our results and discuss open problems and future work.

2 ALGORITHMS UNDER EXAMINATION

The learning-to-rank algorithm, that we enhance in order to perform the experiments are AdaRank (Xu and Li, 2007), RankBoost (Freund, Iyer, Schapire, Singer, 2003) and RankNet (Burges, Shaked, Renshaw, Lazier, Deeds, Hamilton and Hullender, 2005).

RankBoost is a pairwise learning-to-rank algorithm and like all the boosting algorithms it operates in rounds. On each round, RankBoost calls the weak learner with a view to producing a weak ranking. Also, RankBoost holds a distribution, which is selected to accentuate different parts of the training data, which is passed on each round to the weak learner. If a pair of instances is assigned with a high weight, it indicates a great importance that the weak learner orders that pair correctly. The final ranking is a weighted sum of the weak rankings.

AdaRank is a listwise learning-to-rank algorithm and similarly like all the boosting algorithms it operates in rounds. AdaRank uses a training set as input and takes the performance measure function and the number of iterations as parameters. AdaRank runs rounds and at each round, it retains a distribution of weights over the queries in the training data, it creates a weak ranker. Initially, AdaRank defines equal weights to the queries and then at each round it increases the weights of those queries that are not ranked properly. As a result, the learning at the next round concentrates on the generation of a weak ranker that is able to work on the ranking of those 'hard' queries. Finally, it outputs a ranking model by linearly combining the weak rankers. The AdaRank's characteristic attribute is that for the computation of the distribution of the weights over the queries it uses the evaluation of the

documents' labels of the ranked list and not the documents' values directly.

RankNet is a pairwise learning-to-rank algorithm where the loss function, as it is obvious, is defined on a pair of documents, but the hypothesis is defined with the use of a scoring function. The target probability is defined based on the ground truth labels of the given two documents related to a training query. Thereafter, the difference between the scores of these two documents given by the scoring function is used to construct the modelled probability and the cross entropy between the target probability and the modelled probability is used as the loss function. A neural network is used as the model and gradient descent as the optimization algorithm to learn the scoring function.

3 OUR APPROACH

Intuitively, the basic insight behind our idea is centered around the hypothesis that the quality of ranking, which is the result of the learning-to-rank process, can be improved if we take into account the auxiliary information provided by the multi-way inter-relationship between all the documents.

A novel task named "Relational Ranking" (Liu, 2011) for learning-to-rank, apart from the properties of each individual document in the ranking procedure, also makes allowances for the inter-relationship between the documents. The kind of this connection determines the targeted application; for example measures of disjointedness (overlap minimization) are applied to search result diversification, while measures of content similarity for topic extraction/distillation. Generally, the ranked lists are generated by sorting the documents according to their scores output by the learned model. However, it is common sense that in some practical cases we should allow for the relationships between the documents, and it is not adequate to define the scoring function exclusively on discrete documents. The existing works on relational ranking do not only use a matrix or a graph, which must be predefined by experts, to model the relationship, but also are based on pairwise relationship. The pairwise relationship, either similarity, dissimilarity, or preference, is very restrictive and so it is very difficult to use relational ranking in a lot of real applications. For example (Liu, 2011), all the webpages in the same website have a inter-relationship. It is more rational to use a hypergraph to model such inter-relationships.

We try to cope with the above restrictions and to

create a non-predefined structure that illustrates the multi-way inter-relationship between all the documents. This paper has as purpose to present how we can incorporate in an existing learning-to-rank algorithm's function the clustering's structure so as to gain better ranked lists.

The objective of the clustering (Kurland, 2006; Gan, Ma and Wu, 2007) is to separate an unstructured corpus of documents into clusters. We want the documents to be as similar to each other in the same cluster and as dissimilar to documents from other clusters as possible. The *cluster hypothesis* (Kurland, 2013; van Rijsbergen, 1979) states the fundamental assumption we make when using clustering in information retrieval, namely that documents in the same cluster behave similarly with respect to relevance to information needs. So, if there is a document from a cluster that is relevant to a query, then it is likely that other documents from the same cluster are also relevant. Many researchers (Raiber and Kurland, 2012; Hearst and Pedersen, 1996) have depicted that the cluster hypothesis holds on the Web and since clustering has gained great attention, much research (McKeown et al., 2002; Liu, Bruce, 2004) has been done on what are its benefits. So, it states that the users should expect to see similar documents close to each other in a ranked list. Of course, one could argue that cluster hypothesis is valid only if the similarity measure used for the clustering is similar to the content based algorithm used for the query. However it is rational to assume that the provided clustering gathers documents according to their information content. Hence, since information retrieval aims at satisfying information needs, clustering could be useful for the information seeker. Thus, our intention is to make the most of the benefits of the clustering and those of learning-to-rank in order to improve the efficacy in the ranked lists.

The learning-to-rank algorithms are iterative. This attribute helps our approach to gather each document near its cluster's documents at the ranked list gradually during the algorithm's iterations. Our approach is to create a per query clustering and to give to each document, during algorithm's iterations, a bonus proportional to the cluster in which it belongs to. So, we estimate that with the passage of iterations similar documents will appear together, since we promote similar documents with similar bonus, and particularly the documents, which belong to the cluster that has the centroid with the best BM25 (Manning, Raghavan, Schutze, 2008) value, will be at the top of the ranked list as they are the documents that get the greatest bonus. With this

process, we regard that there should be a uniform classification where the documents will be displayed in descending order according to their labels.

With the above-mentioned, we expect that we will take better evaluations according with the performance measures such as MAP, NDCG@k and P@k (Baeza-Yates and Ribeiro-Neto, 2011).

Our conviction that through the above process we will take better retrieval metrics is based on the assumption that the cluster, which has the centroid with the best BM25 value, will contain the documents that have the best label and consequently are the most relevant. So, through the iterations this cluster will be appeared at the top of the ranked list and as a consequence the documents with the best label, will appear at the top of the ranked list respectively. Therefore, we will get better performance measures. Here is an example of ranked lists, where the numbers 4, 3, 2, 1, 0 are the documents' labels and the number 4 indicates the best relevance and the number 0 indicates the irrelevance, which illustrates graphically our goal. We should also mention that each of the number (0,1,2,3,4) indicates a distinguished cluster and each document belongs to the cluster of its label:

| | default | our conviction |
|--------------|---------|----------------|
| 1st result: | 4 | 4 |
| 2nd result: | 3 | 4 |
| 3rd result: | 4 | 3 |
| 4th result: | 1 | 3 |
| 5th result: | 2 | 2 |
| 6th result: | 3 | 2 |
| 7th result: | 2 | 2 |
| 8th result: | 1 | 1 |
| 9th result: | 2 | 1 |
| 10th result: | 0 | 0 |

As default we consider a ranked list that has been generated by a learned model based on the single documents. The above example depicts how we want to muster each cluster's documents together and promote the best clusters with the best relevance labels at the top of the ranked list. It is obvious that according to our conviction we get better performance metrics.

A main framework for a learning-to-rank algorithm, which operates according to our approach, would be the following:

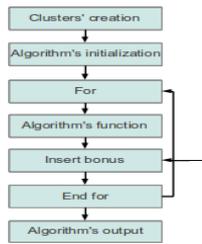


Figure 1: Learning-to-Rank algorithm's framework using our approach.

Observing the above framework we notice that the innovative idea, which stands out from the common learning-to-rank algorithms, is the clusters' creation and the bonus insertion as the algorithms' function is intact.

For all of our experiments, we chose that the given bonus should contain the BM25 value. We made this choice, because the BM25 value has been used quite widely and quite successfully across a wide range of experiments and it has been shown to be the best of the known probabilistic weighting schemes. Furthermore, it is evident that the BM25 value can completely depict the degree of correlation between the cluster and the query.

As bonus for AdaRank-CC algorithm we use the product $(b/s)*f(x)$ where b is the BM25 value of the cluster's centroid in which the document belongs to, s is the sum of the bm25 values of the clusters' centroids that correspond to the specific query and $f(x)$ is the document's value from the algorithm.

We decided to divide the BM25 value with the s value so as to give to each document a normalized bonus in relation to the other clusters' BM25 values.

So, before the algorithm starts we create the clustering and at the end of each iteration, after the algorithm's function is complete, we update the value of each document as following

$$f(x) = f(x) + ((b/s) * f(x)) \tag{1}$$

As bonus for the RankBoost-CC algorithm we have experimented with many values, following the same reasoning as before, but none of these improved the efficiency of RankBoost algorithm. So, we did not get an indicative type of bonus. However, the most successful formula was $(b/s)*f(x)$ where b is the BM25 value of the cluster's centroid in which the document belongs to, s is the sum of the bm25 values of the clusters' centroids that correspond to the specific query and $f(x)$ is the document's value from the algorithm. The values are the same as in AdaRank-CC algorithm, but without having the desired results.

In the following we present the AdaRank-CC

and RankBoost-CC algorithms which follow the same philosophy.

AdaRank-CC/RankBoost-CC Algorithm:

Clustering:clusters' creation
 Initialization:AdaRank's/RankBoost's initialization
 For
 ^ AdaRank's/RankBoost's function
 ^ For each document

- o Find the cluster in which document belongs to and get its BM25 value
- o Update the value of the document using the above BM25 value

 ^ End for
 End for
 Output:AdaRank-CC's/RankBoost-CC's output

As bonus for the RankNet-CC algorithm we use $(b/10^4)*f_{value}(x)$ where b is the BM25 value of the cluster's centroid in which the document belongs to and $f_{value}(x)$ is a document's feature value from the algorithm. At this algorithm, we use the documents' vectors updating their feature values at each iteration instead of the documents' values as we did before.

We decided to divide the BM25 value with the number 10000, because through the experiments we got the best results.

So, before the algorithm starts we create the clustering and at the end of each iteration, after the algorithm's function is complete, we update the elements of the documents' vectors as follows:

$$f_{value}(x) = f_{value}(x) + ((b/10^4) * f_{value}(x)) \tag{2}$$

The RankNet-CC algorithm follows the AdaRank-CC's and RankBoost-CC's philosophy, but, instead of updating the documents' value, it updates each element of the documents' feature vector.

So in contrast to the AdaRank-CC and RankBoost-CC algorithms, the RankNet-CC algorithm, based on the theory that better feature vectors provide better results, tries to update the documents' feature vectors at each iteration, promoting the documents that belong to the clusters with the best BM25 value. With the above-mentioned, at each iteration we provide better feature values at the documents' vectors, which belong to the best clusters, targeting the neural network to provide better values to these documents.

4 EXPERIMENTAL EVALUATION

We conducted experiments to investigate the performance of our implementations using the two Microsoft Learning to Rank Datasets (<http://research.microsoft.com/en-us/projects/mslr/>). Also, for our experiments we used the RankLib (<http://people.cs.umass.edu/~vdang/ranklib.html>) library, which contains eight popular learning-to-rank algorithms and many retrieval metrics.

These two datasets are machine learning data and they consist of feature vectors exported from query-url pairs in company with relevance judgment labels. The queries and urls are represented by IDs. Also, each query-url pair is represented by a 136-dimensional vector, in which every dimension provide some information. In order to create our clustering, we have chosen 24 specific features, which we consider as more informative, so as to create a better clustering. We have selected the features 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 85, 90, 95, 110, 130, 133, 134, 136 that correspond to the whole document's covered query term number, covered query term ratio, stream length, IDF(Inverse document frequency), sum of term frequency, min of term frequency, max of term frequency, variance of term frequency, sum of stream length normalized term frequency, min of stream length normalized term frequency, max of stream length normalized term frequency, mean of stream length normalized term frequency, variance of stream length normalized term frequency, sum of $tf*idf$, min of $tf*idf$, max of $tf*idf$, mean of $tf*idf$, variance of $tf*idf$, BM25, PageRank, QualityScore2, Query-url click count and url dwell time respectively.

The purpose of our experiments was to depict the usefulness of exploiting cluster information in Learning-to-Rank. We have created a per query clustering using the algorithm k-means++, which is a variant of the k-means algorithm (Gan, Ma and Wu, 2007) for choosing the initial values (or "seeds") for the implementation of the algorithm. In the assignment step of the k-means++ algorithm, each document was assigned to the cluster whose mean was the "nearest" to it according to the squared Euclidean distance. We chose euclidean measure and the specific set of features so that documents in the same cluster can have similar characteristics concerning the various anticipated information needs. We should also note that every dataset has a variable number of documents that correspond to a specific query. Hence, we have

queries that have for example 5 results and others that have 40 results. For this reason, we have queries that have from 2 to 5 clusters, depending on the number of their documents.

As we will see, in the presentation of the experiments, though our approach aims at evaluation effectiveness it also comes as an extra bonus an improvement in efficiency.

4.1 Experiments with MSLR-WEB10K

In this experiment, we made use of the MSLR-WEB10K data to test the performance of AdaRank, AdaRank-CC, RankNet and RankNet-CC. The MSLR-WEB10K consists of 10,000 queries and is partitioned into 5 folders.

The following table shows the difference in the value of metrics, based on the average of the five folders, between AdaRank and AdaRank-CC.

Table 1: Comparison between AdaRank and AdaRank-CC.

| | AdaRank | AdaRank-CC |
|---------|---------|------------|
| NDCG@3 | 0,36562 | 0,36758 |
| NDCG@5 | 0,31002 | 0,34954 |
| NDCG@10 | 0,34352 | 0,39596 |
| P@3 | 0,69476 | 0,69438 |
| P@5 | 0,66332 | 0,66174 |
| P@10 | 0,59406 | 0,62542 |
| MAP | 0,57236 | 0,57622 |

The following table shows the difference in iterations, based on the average of the five folders, between AdaRank and AdaRank-CC.

Table 2: Comparison between AdaRank and AdaRank-CC.

| | AdaRank | AdaRank-CC |
|---------|---------|------------|
| NDCG@3 | 54,4 | 39,8 |
| NDCG@5 | 119,2 | 59,4 |
| NDCG@10 | 95 | 35 |
| P@3 | 10,6 | 8,6 |
| P@5 | 7,8 | 7,6 |
| P@10 | 122,2 | 43,8 |
| MAP | 163 | 62 |

The following table shows the difference in the value of metrics, based on the average of the five folders, between RankNet and RankNet-CC.

4.2 Experiments with MSLR-WEB30K

In this experiment, we made use of the MSLR-WEB30K data to test the performance of AdaRank, AdaRank-CC, RankNet and RankNet-CC. The

Table 3: Comparison between RankNet and RankNet -CC.

| | RankNet | RankNet -CC |
|---------|---------|-------------|
| NDCG@3 | 0,1573 | 0,1531 |
| NDCG@5 | 0,1683 | 0,1665 |
| NDCG@10 | 0,2002 | 0,2038 |
| P@3 | 0,4716 | 0,4711 |
| P@5 | 0,4480 | 0,4410 |
| P@10 | 0,4431 | 0,4415 |
| MAP | 0,4421 | 0,4446 |

MSLR-WEB30K consists of 30,000 queries and is partitioned into 5 folders.

The following table shows the difference in the value of metrics, based on the average of the five folders, between AdaRank and AdaRank-CC.

Table 4: Comparison between AdaRank and AdaRank-CC.

| | AdaRank | AdaRank-CC |
|---------|---------|------------|
| NDCG@3 | 0,38562 | 0,34059 |
| NDCG@5 | 0,30028 | 0,33694 |
| NDCG@10 | 0,34796 | 0,39516 |
| P@3 | 0,69632 | 0,69736 |
| P@5 | 0,66686 | 0,66588 |
| P@10 | 0,60698 | 0,63182 |
| MAP | 0,57822 | 0,58574 |

The following table shows the difference in iterations, based on the average of the five folders, between AdaRank and AdaRank-CC.

Table 5: Comparison between AdaRank and AdaRank-CC.

| | AdaRank | AdaRank-CC |
|---------|---------|------------|
| NDCG@3 | 39,2 | 64,8 |
| NDCG@5 | 110,2 | 62,6 |
| NDCG@10 | 84,8 | 31 |
| P@3 | 9,1 | 8 |
| P@5 | 18,8 | 6,8 |
| P@10 | 86,6 | 46,6 |
| MAP | 169 | 51,8 |

The following table shows the difference in the value of metrics, based on the average of the five folders, between RankNet and RankNet-CC.

Table 6: Comparison between RankNet and RankNet -CC.

| | RankNet | RankNet -CC |
|---------|---------|-------------|
| NDCG@3 | 0,1558 | 0,1593 |
| NDCG@5 | 0,1686 | 0,1690 |
| NDCG@10 | 0,2019 | 0,2043 |
| P@3 | 0,4706 | 0,4718 |
| P@5 | 0,4475 | 0,4433 |
| P@10 | 0,4422 | 0,4410 |
| MAP | 0,4435 | 0,4467 |

4.3 Inference from the Experiments

Regarding the AdaRank-CC, which is an algorithm that doesn't use directly the documents' values with the additional bonus in its function, is that exploiting the clustering and the bonus to each document during the iterations, we can get better results considering the NDCG@k, MAP and P@k metrics simultaneously in fewer iterations. More precisely, observing the graphs we understand that for NDCG@3 and P@3 we have approximately the same results between the default AdaRank and AdaRank-CC. But, for NDCG@5, P@5 and especially for NDCG@10, P@10 and MAP we observe that the AdaRank-CC provides better results. This observation confirms our conviction that through the bonus during the iterations we will direct the documents of the best clusters at the top of the ranked list and this also shows that we gather the documents with the best labels at the top 10 positions and as result we have better evaluation.

Hence, we conclude that our approach to combine learning-to-rank with an existing clustering can be integrated with positive results in fewer iterations at an algorithm such as the AdaRank which is positively affected by the additional bonus that are given to the documents. We infer this algorithm's improvement to the additional bonus observing the calculation of distribution at each iteration. The distribution's calculation is the following (Li, 2011):

$$P_{i+1} = \frac{\exp(-E(\pi_i, y_i))}{\sum_{j=1}^m \exp(-E(\pi_j, y_j))} \quad (3)$$

where $E(\pi, y)$ is the evaluation conducted at the list level, t is the number of iteration, π is the ranked list of documents, y is the list of documents' labels and i is the number of query.

So, the distribution's calculation is based on the evaluation of the documents' labels and not on the documents' values, given by the scoring function of the algorithm, in which we put the additional bonus.

In contrast to the above conclusions, regarding the Rank-Boost-CC, for which the documents' values have an important role in algorithm's distribution determination, we don't get better evaluation. More precisely, we can understand the effect of the documents' value, observing how the distribution is calculated. At each iteration the distribution is calculated using this formula (Liu, 2011):

$$D_{t+1}(x_0, x_1) = \frac{D_t(x_0, x_1) \exp(a_t(h_t(x_0) - h_t(x_1)))}{Z_t} \quad (4)$$

where Z_t is a normalization factor, t is the number of iteration, x is a document, a_t is a parameter and $h(x)$ is the document's value.

So, the distribution's calculation is based on the documents' values which contain the additional bonus. It is clear that, in contrast to the AdaRank as it uses the documents' labels evaluation, the documents' value plays a significant role to the distribution's value. Since, we don't get better result using the additional bonus for this kind of distribution calculation, we can deduce that the additional bonus adversely affects these algorithms such as RankBoost as it adversely distorts the calculation of the distribution.

Regarding the RankNet-CC, for which at the end of each iteration we update the elements of the documents' vectors in order to create better vectors and as consequence better results, from the results we can observe that the metrics between RankNet and RankNet-CC are approximately equal and so we can not infer reliable conclusions. Slightly better results in favour of RankNet-CC we can observe for the metrics NDCG@10 and P@10 and this remark agrees with the observation that we made for the AdaRank-CC concerning the above two metrics.

5 CONCLUSIONS

In this paper we have proposed new versions of the AdaRank, RankBoost and RankNet learning to rank algorithms, referred to as AdaRank-CC, RankBoost-CC and RankNet-CC respectively. In contrast to existing methods, AdaRank-CC, RankBoost-CC and RankNet-CC take into consideration the multi-way inter-relationship between all documents, since we have separated the unstructured set of documents into clusters using the k-Means++ algorithm.

Our basic finding in this work is that algorithms such as AdaRank-CC, for which the additional bonus doesn't affect the computation of the distribution of weights over the queries, can indeed improve both effectiveness and efficiency, as we can get better overall quality according to the well known evaluation metrics (NDCG, MAP, various levels of precision) and simultaneously decrease the number of iterations. As future work, it could be interesting to further investigate how we can get the similar results to those of the AdaRank-CC and the other algorithms that use directly the documents' values with the additional bonus in their function and consequently they are affected by them.

ACKNOWLEDGEMENTS

This research has been co-financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Program: Thales. Investing in knowledge society through the European Social Fund.

REFERENCES

- Baeza-Yates R., and Ribeiro-Neto B., (2011) Modern Information Retrieval: the concepts and technology behind search. *Addison Wesley*, Essex.
- Burges C., Shaked T., Renshaw E., Lazier A., Deeds M., Hamilton N. and Hullender G., (2005) *Learning to Rank using Gradient Descent*, ICML 2005: 89-96.
- Freund Y., Iyer R., Schapire R. E, Singer Y., An Efficient Boosting Algorithm for Combining Preferences. *In Journal of Machine Learning Research* 4 (2003), 933-969.
- Gan G., Ma C. and Wu J. (2007). *Data Clustering: Theory, Algorithms, and Applications*. DOI=<http://dx.doi.org/10.1137/1.9780898718348>.
- Hearst A. M., Pedersen J. O., Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results, *In Proceedings of ACM SIGIR '96*, August, 1996, Zurich.
- Kurland O., Inter-Document similarities, language models, and ad-hoc information retrieval. Ph.D. Thesis (2006).
- Kurland O., The Cluster Hypothesis in Information Retrieval, SIGIR 2013 tutorial (2013). <http://iew3.technion.ac.il/~kurland/clustHypothesisTutorial.pdf>.
- Li H., Learning to Rank for Information Retrieval and Natural Language Processing. (2011) *Morgan & Claypool*.
- Liu T. Y., Learning to Rank for Information Retrieval. (2011) *Springer*.
- Liu, X, and W. Bruce C. 2004. Cluster-based retrieval using language models. In Proc. SIGIR, pp. 186-193. ACM Press. DOI: [doi.acm.org/10.1145/1008992.1009026](https://doi.org/10.1145/1008992.1009026).
- Manning C. D., Raghavan P., Schütze H., (2008) Introduction to Information Retrieval, Cambridge University Press, pp. 232-234.
- McKeown et al. (2002), Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster, In Proc. Human Language Technology Conference.
- Raiber F., Kurland O. (2012), Exploring the Cluster Hypothesis, and Cluster-Based Retrieval, over the Web, ACM CIKM: 2507-2510.
- Robertson, S., Zaragoza, H., Taylor, M. (2004) Simple BM25 extension to multiple weighted fields.. In CIKM 2004: *Proceedings of the thirteenth ACM International Conference on Information and Knowledge Management*, pages 42-49.

- van Rijsbergen, C. J.: *Information Retrieval, 2nd edn., Butterworths (1979).*
- Xu J. and Li H., (2007) *AdaRank: A Boosting Algorithm for Information Retrieval*, SIGIR 2007: 391-398.
- Zeng H.-J., He Q.-C., Chen Z., Ma W.-Y., Ma J. (2004), *Learning to Cluster Web Search Results*. SIGIR 2004: 210-21.

Automated Identification of Web Queries using Search Type Patterns

Alaa Mohasseb¹, Maged El-Sayed² and Khaled Mahar¹

¹*College of Computing & Information Technology, Arab Academy for Science, Technology & Maritime Transport, Alexandria, Egypt*

²*Department of Information Systems & Computers, Alexandria University, Alexandria, Egypt
{alaamohasseb, khmahar}@aast.edu, maged@alexu.edu.eg*

Keywords: Information Retrieval, User Intent, Web Queries, Web Searching, Search Engines, Query Classification.

Abstract: The process of searching and obtaining information relevant to the information needed have become increasingly challenging. A broad range of web queries classification techniques have been proposed to help in understanding the actual intent behind a web search. In this research, we are introducing a new solution to automatically identify and classify the user's queries intent by using Search Type Patterns. Our solution takes into consideration query structure along with query terms. Experiments show that our approach has a high level of accuracy in identifying different search types.

1 INTRODUCTION

The main goal of any information retrieval system is to obtain information relevant to information needs. Search engines can better help the user to find his/her needs if they can understand the intent of the user. Identifying such intent remains very difficult; one major task in identifying the intent of the search engine users is the classification of the query type.

There are many different proposed classifications of web queries (Morrison, et al., 2001, Broder, 2002, Kellar, et al., 2006, Baeza-yates, et al., 2006, Ashkan, et al., 2009, Lewandowski, et al., 2012, Bhatia, et al., 2012). Broder's classification of web queries (Broder, 2002) is one of the most commonly used classifications. It classifies web queries to three main types: Informational queries, Navigational queries and Transactional queries.

Some researches (Choo, et al. 2000, Morrison, et al., 2001, Broder, 2002, Rose, et al., 2004, Kellar, et al., 2006) used different manual methods to classify users' queries like surveys and field studies. Other researches used automated classification techniques like supervised learning, SVM...etc. (Lee, et al., 2005, Beitzel, et al., 2005, Baeza-yates, et al., 2006, Liu, et al., 2006, Ashkan, et al., 2009, Mendoza, et al., 2009, Jansen, et al., 2010, Kathuria, et al., 2010).

One drawback of the solutions that were introduced so far is that they do not take into consideration the structure of the queries. Queries submitted to search engines are usually short and

ambiguous and most of the queries might have more than one meaning, therefore using only the terms to identify search intents is not enough, two queries might have exactly the same set of terms but may reflect two totally different intents, therefore classifying web queries using the structure of the query in addition to terms and characteristics may help in making the classification of queries more accurate.

In our research, we propose a solution that automatically identifies and classifies user's queries using Search Type Patterns. Such Search Type Patterns are created from studying different web queries classification proposals and from the examination of various web logs. A Web Search Pattern is constructed from one or more terms, such terms are categorised and introduced in the form of taxonomy of search query terms.

We have developed a prototype to test the accuracy of our solution. Experimental results show that our solution accurately identified different search types.

The rest of the paper is organized as follows: Section 2 highlights the different proposed classification techniques used in web query identification. Section 3 provides detailed explanation of the extended classification of web search queries and the different type of each category. Section 4 provides a detailed description of the proposed solution. Section 5 covers experiments and results and finally Section 6 gives

conclusion and future work.

2 PREVIOUS WORK

2.1 Search Types

According to (Broder, 2002) web searches could be classified according to user's intent into three categories: Navigational, Informational and Transactional. Many researches (Liu, et al., 2006, Jansen, et al., 2008, 2010, Mendoza, et al., 2009, Kathuria, et al., 2010, Hernandez, et al., 2012) have based their work on Broder's classification of user query intent. Others like (Lee, et al., 2005) used navigational and informational queries only due to lack of consensus on transactional query and to make classification task more manageable.

Rose, et al., (2004) and (Jansen, et al., 2008) extended the classification of Informational, Navigational and Transactional queries by adding level two and level three sub-categories.

Lewandowski, et al., (2012) proposed two new query intents, Commercial and Local. According to their work, the query might have a Commercial potential like the query: "*commercial offering*" or the user might search for information near his current location.

Bhatia, et al., (2012) classified queries to four classes: Ambiguous, Unambiguous but Underspecified, Information Browsing and Miscellaneous.

Calderon-Benavides, et al., (2010) and Ashkan, et al., (2009) proposed other classification of queries that classified user intent into dimensions and facets. These dimensions and facets are extracted from user's queries to help the identification of user intent when searching for information on the web like Genre, objective, specificity, scope, topic, task, authority sensitivity, spatial sensitivity and time sensitivity (Calderon-Benavides, et al., 2010).

Ashkan, et al., (2009) classified query intent into two dimensions, Commercial and Non-commercial and Navigational and Informational.

Kellar, et al., (2006) classified web informational task based on three main informational goals, Information Seeking, Information Exchange and Information Maintenance.

Baeza-yates, et al., (2006) established three categories for user search goal, Informational, Not Informational and Ambiguous. Informational query when the user's interest is to obtain information available on the web. Not Informational include specific transactions or resources like "*buy*",

"*download*"...etc. Ambiguous queries include queries that can't be identified directly because the user interest is not clear.

Morrison, et al., (2001) classified search goals into Find, Explore, Monitoring and Collect, this classification focus on three variables: the purpose of the search, the method used to find information and the contents of searched information.

2.2 Classification Methods and Techniques

Researchers have used different manual and automated classification methods and techniques to identify users query intent.

Broder, (2002) classified user's query manually by using a survey of AltaVista users as one of the methods to determine the type of queries, the survey was done online and users were selected randomly. Users were asked to describe the purpose of their search, queries that were neither Transactional nor Navigational were assumed to be Informational, the final results of the survey showed that 24.5% of the queries were Navigational, Informational queries accounted for 39% of the queries and transactional accounted for 36% of the queries. In addition Broder has analysed a random set of 1000 queries from the daily AltaVista log, queries that were neither Transactional nor Navigational were assumed to be Informational, results showed that 20% of queries were Navigational, 48% were Informational and 30% were Transactional.

Choo, et al., (2000) and Kellar, et al., (2006) used questionnaire survey for manual classification of queries and since participants in this kind of classification were low in number, the results can't be considered reliable.

In addition to the questionnaire survey (Kellar, et al., 2006) conducted one-week field study to classify data using a custom web browsing and analysed the data for only 21 participants.

Rose, et al., (2004) argued that user goals can be deduced from looking at user behaviour available to the search engine like the query itself, result clicked...etc. This approach has limitation that the goal-inferred from the query may not be the user actual goal.

Lewandowski, et al., (2012) analysed click-through data to determine Commercial and Navigational queries and used crowdsourcing approach to classify a large number of search queries.

Liu, et al., (2006) also used click-through data for query type identification. Queries were randomly

selected and manually classified by three assessors using voting to decide queries category. This work relied on decision tree algorithm and used precision and recall to test effectiveness of the query type identification.

Lee, et al., (2005) proposed two types of features, past user click behaviour and Anchor-link distribution. Results showed that the combination of these two techniques could correctly identify the goals for 90% of the queries.

Hernandez, et al., (2012) introduced a solution that automatically classifies queries using only the text included in the query, based on the feature and characteristics described by (Broder 2002, Jansen, et al., 2008, Dayong, et al., 2010). More than 1692 queries were manually classified then two machine-learning algorithms, naïve Bayes and Support Vector Machine (SVM), were used. Results showed that the two machine-learning algorithms suited more Informational and Transactional queries; results of Navigational queries were very low with naïve Bayes and null with SVM. These Results indicate that using only the content of words in the queries is not sufficient to find all user intents.

Ashkan, et al., (2009) classified 1700 queries and manually labelled the selected queries then used ads click-through and query features to determine the query intent.

Beitzel, et al., (2005) and Baeza-Yates, et al., (2006) used supervised learning to determine query intents. In addition to supervised learning (Baeza-Yates, et al., 2006) applied unsupervised learning then combined both techniques to identify user search goal.

Jansen, et al., (2008) developed a software application that automatically classified queries using web search engine log of over a million and a half queries. Results showed that more than 80% of web queries were Informational, Navigational and Transactional queries each represent about 10% of web queries. To validate their approach 400 queries from Dogpile transaction log were randomly selected and manually coded, 74% of the queries were successfully classified and the remaining 25% were vague or multi-faceted queries.

Kathuria, et al., (2010) automatically classified queries using k-means clustering, results for this technique showed that more than 75% of web queries are Informational in nature and 12% each for navigational and transactional queries.

3 BACKGROUND

3.1 Web Search Queries Classification

The following sections describe in details each of the categories we considered in our work. These categories are based on work done by (Broder, 2002, Rose, et al., 2004 and Jansen, et al., 2008).

3.1.1 Informational Searching

Informational Searching has five sub-categories:

a) Informational - Directed (I, D): the goal of this category is to learn something in particular about a certain topic, or to answer a specific question, both open and closed ended. This category has level two sub-categories:

a.1) Informational - Directed - Open (I, D, O): this category may take many forms either a question to get an answer for an open-ended question or one with unconstrained depth or to find information about two or more topics. Examples: "*why are metals shiny?*" and "*honeybee communication*".

a.2) Informational - Directed - Closed (I, D, C): queries in this category can be a question to find one specific or unambiguous answer or to find information about one specific topic. Examples: "*capital of Brazil*" and "*what is a prime number?*"

b) Informational - Undirected (I, U): the purpose of this category is to know anything and everything about a topic, most queries in this type are related to science, medicine, history and news and celebrities (Rose, et al., 2004). Examples: "*Shawn Johnson*", "*Vietnam war*" and "*hypertension*".

c) Informational - List (I, L): plural query terms are a highly reliable indicator of this category (Rose, et al., 2004), the goal of this type of queries is to find a list of suggested websites or candidates or list of suggestions for further research. Examples: "*list of Disney movies*", "*London universities*", and "*things to do in Atlanta*".

d) Informational - Find (I, F): the goal of this category is to find or locate something in the real world like a product or service. Most product or shopping queries have the locate goal (Rose, et al., 2004), for example: "*apple store location in New Jersey*" and "*cheap apple MacBook pro*".

e) Informational - Advice (I, A): the goal of this category is to get ideas, suggestions, advice or instructions about something and may take many forms like a question. Examples: "*How to download iTunes*" and "*writing a book*".

3.1.2 Navigational Searching

Navigational Searching has two sub-categories:

a) Navigational to Transactional (N, T): the URL or website user is searching for is a transactional site. Examples: "*amazon.com*" and "*ebay.com*".

b) Navigational to Informational (N, I): the URL or website user is searching for is an informational site. Examples: "*google.com*" and "*yahoo.com*".

3.1.3 Transactional Searching

Transactional Searching has the following sub-categories:

a) Transactional - Obtain (T, O): the goal of this type of queries is to obtain specific resource or object, not to learn some information but just to use the resource itself. This category has the following level two sub-categories:

a.1) Transactional - Obtain - Online (T, O, O): the resources of this type of queries will be obtained online, meaning that the user might search for something to just look at it on the screen. Examples: "*meatloaf recipes*" and "*Adele Songs lyrics*".

a.2) Transactional - Obtain - Offline (T, O, F): the resources of this type of queries will be obtained offline and may require additional actions by the user, meaning that the user might search for something to print or save to use it later offline. Examples: "*Bon Jovi wallpapers*" and "*windows 7 screensavers*".

b) Transactional - Download (T, D): the resource of this type of query is something that needs to be installed on a computer or other electronic device to be useful like finding a file to download. This category has level two sub-categories:

b.1) Transactional - Download - Free (T, D, F): the downloadable file is free. Examples: "*free online games*" and "*free mp3 downloads*".

b.2) Transactional - Download - Not Free (T, D, N): the downloadable file is not necessarily free. Examples: "*safe haven book download*" and "*Kelly Clarkson songs download*".

c) Transactional - Interact (T, I): this type of queries occurs when the intended result of the search is a dynamic web service, and requires further interaction with a program or a resource. Examples: "*currency converter*", "*stock quote*", "*buy cell phones*", and "*weather*".

d) Transactional - Results Page (T, R): the goal of this category is to obtain a resource that can be printed, saved, or read from the search engine

results page. This category has level two sub-categories:

d.1) Transactional - Results Page - Links (T, R, L): the resources of this kind of queries appear in the title, summary, or URL of the search engine results page. Example: "*searching for title of a conference paper to locate the page numbers*".

d.2) Transactional - Results Page - Other (T, R, O): the resources of this kind of queries does not appear on the search engine results page but somewhere else on the search engine results page. Example: "*spelling check of a certain term*".

3.2 Characteristics of Web Search Queries

3.2.1 Informational Search Characteristics

One of the major characteristics of Informational Searching is the use of natural language phrases (Jansen, et al., 2008). Queries for such search may consist of informational terms like "*list*" and "*playlist*"...etc., question words like "*who*", "*what*", "*when*"...etc. Searches related to Advice, help and guidelines like "*FAQs*" or "*how to*"...etc., ideas and suggestions terms, recent information and news like "*weather*".

Some queries consisting of multimedia like videos are considered informational like "*how-to-do*" videos. Topics related to science, medicine, history, news and celebrities are also considered informational, (Rose, et al., 2004).

3.2.2 Navigational Search Characteristics

Navigational Searching queries contain organization, business, company and universities name, domain suffixes like "*.com*", "*.org*"...etc. also prefixes such as "*www*" or "*http*" and "*web*" as the source. Some Navigational queries contain URLs or parts of URLs (Jansen, et al., 2008).

Most queries consisting of people names, including celebrities, are not considered navigational. According to (Rose, et al., 2004) a search for a celebrity such as "*Justin Timberlake*" will result in a fan or media sites, and usually the goal or objective of searching for a celebrity is not just visiting a specific site.

3.2.3 Transactional Search Characteristics

According to (Jansen, et al., 2008) queries in Transactional Searching is related to obtaining terms like "*lyrics*", "*recipes*", "*patterns*"...etc., download terms like, "*software*"...etc. Also queries containing

"audio", "video" and "images" are considered to be transactional.

Queries related to entertainment terms like "pictures", "games"...etc., and e-commerce. Interact terms such as "buy", "chat", "book", "order"...etc., and file extensions like "jpeg", "zip...etc.", (Jansen, et al., 2008).

4 PROPOSED SOLUTION

Our solution mainly relies on Search Type Patterns (STPs). These patterns generalize web search queries of different types and could be used in identifying the query class and hence the user's intent. We have constructed 1182 different Search Type Patterns. Examples of these patterns are given in sections 4.1 and 4.2. Due to space limitation we couldn't give a comprehensive listing of these patterns.

Our proposed Search Type Patterns cover all categories discussed in section 3.1 above except Navigational search sub-categories and the Transactional-Results page category. The reason of excluding these categories is because it is not possible to determine the intent of the query without performing the search and monitoring the user's interaction with the result, which falls outside the scope of our work since our solution is not based on processing the search results. For example, if a user searches for: "UCLA University", he might be interested in browsing the site to know more information (Navigational-to-Informational) or to register a course (Navigational-to-Transactional).

Each Search Type Pattern (STP) is composed of a sequence of term categories (tc). $STP = \langle tc_1, tc_2, \dots, tc_n \rangle$. Each term category tc_i contains a list of terms. The categorization of terms in our solution is mainly based on the seven major word classes in English: Verb, Noun, Determiner, Adjective, Adverb, Preposition and Conjunction. In addition to that we added a category for question words that contains the six main question words: How, who, when, where, what and which. We further extended this classification by adding two super-categories: Domain Suffixes and Prefixes. We also added sub-categories where a category may have one or more sub-categories.

Term sub-categorization is built in a way that enables the preservation of uniqueness of each Search Type Pattern. In other words, no two Search Type Patterns will have exactly the same sequence of term categories. Section 4.1 will discuss in details

how term categorization and Search Type Patterns were constructed.

Table 1 shows detail of all term categories in our solution and Figure 1 shows the taxonomy organization of these categories.

Table 1: List of Term Categories.

| Category Name | Abbreviation | Terms |
|---|--------------|---|
| Action Verb-Interact terms | AV_I | Buy, Reserve, Order...etc. |
| Action Verb-Locate | AV_L | Locate, Find. |
| Action Verb-Locate & Interact terms | AV_IL | All Locate & Interact terms. |
| Action Verb-Download | AV_D | Download |
| Action Verbs | AV | Write, create, drive...etc. |
| Auxiliary Verb | AuxV | Can, may, will...etc. |
| Linking Verbs | LV | Is, are, was...etc. |
| Verbs | V | All Verbs |
| Adjective Free | Adj_F | Free |
| Adjective Online | Adj_O | Online |
| Adjective Free & Online | Adj_OF | Free & Online |
| Adjective | Adj | All Adjectives |
| Adverb | Adv | Almost, barely, highly...etc. |
| Determiners | D | A, An, The...etc. |
| Conjunction | Conj | And, as, but...etc. |
| Ordinal Numbers | NN_O | 1st, second, 70th...etc. |
| Cardinal Numbers | NN_C | 1, 50, ten...etc. |
| Numeral Numbers | NN | All numbers |
| Celebrities Name | PN_C | Phil Collins, Clint Eastwood, The Beatles...etc. |
| Entertainment | PN_Ent | Specific name of a song, movie, game...etc. |
| Newspapers, Magazines, Documents, Books...etc. | PN_BDN | Specific name of a Newspapers, Magazines, Documents, Books...etc. |
| Events | PN_E | Cannes film festival...etc. |
| Celebrities, Events, Newspapers, Entertainment...etc. | PN_BCEE | All PN_C, PN_BDN, PN_Ent & PN_E |
| Companies Name | PN_CO | IBM, Microsoft, Intel...etc. |
| Geographical Areas | PN_G | London, Europe, Nile River...etc. |
| Places and Buildings | PN_PB | Eiffel Tower, National park...etc. |

Table 1: List of Term Categories. (Cont.)

| Category Name | Abbreviation | Terms |
|---|--------------|---|
| Institutions, Associations, Clubs, Parties, Foundations and Organizations | PN_IOG | Yale university, Warren middle school...etc. |
| Companies, Geographical Areas, Institutions, Places...etc. | PN_CGIP | All PN_CO, PN_G, PN_PB & PN_IOG |
| Celebrities, Entertainment, Companies...etc. | PN_BCC | All PN_BCEE & PN_CGIP |
| Brand Names | PN_BN | Coach, Pepsi, Gucci...etc. |
| Software & Applications | PN_SA | uTorrent, Photoshop, Skype...etc. |
| Products | PN_P | iPad, Oreo cookie...etc. |
| Brand, Products, Software...etc. | PN_BSP | All PN_BN, PN_P and PN_SA |
| Brand, Products, Entertainment, Companies...etc. | PN_BBC | All PN_BCC & PN_BSP |
| History and News | PN_HN | Revolutionary war, American Civil war...etc. |
| Religious Terms | PN_R | Christian, Muslim, God, Allah...etc. |
| Holidays, Days, Months | PN_HMD | Christmas, Saturday, November...etc. |
| Religious Terms, Holidays, Days, Months | PN_HR | All PN_R & PN_HMD |
| Health Terms | PN_HLT | Specific Terms related to health & medicine. |
| Science Terms | PN_S | Specific Terms related to Science. |
| Health & Science Terms | PN_HS | All PN_S & PN_HLT |
| Proper Noun | PN | All Proper Nouns |
| Database and Servers | CN_DBS | Weather, Dictionary...etc. |
| Advice | CN_A | Advice, ideas, instruction, suggestion, tips. |
| Download | CN_D | Download, Software |
| Entertainment | CN_Ent | Music, Movie, Sport, Picture, Game...etc. |
| File Type | CN_File | MP3, PDF...etc. |
| Informational Terms | CN_IFT | List, Playlist...etc. |

Table 1: List of Term Categories. (Cont.)

| Category Name | Abbreviation | Terms |
|---|--------------|---|
| Info. Terms, File & Entertainment | CN_EFI | All CN_Ent, CN_File & CN_IFT |
| Obtain Offline | CN_OF | Wallpapers, documents...etc. |
| Obtain Online | CN_OO | Lyrics, Recipes...etc. |
| Obtain | CN_OB | Obtain Online & Offline |
| File, Entertainment, Informational & Obtain Terms | CN_OBEF | All CN_EFI & CN_OB |
| History & News | CN_HN | History, News, War, Rumour. |
| Interact terms | CN_I | Translation, reservation...etc. |
| Locate | CN_L | Location |
| Site, Website, URL | CN_SWU | Site, Website, URL, Webpage. |
| Common Noun – Other- Singular | CN_OS | All singular common nouns |
| Common Noun- Other- Plural | CN_OP | All plural common nouns |
| Common Noun- Other | CN_O | Other Common Nouns |
| Common Noun | CN | All Common Nouns |
| Pronoun | Pron. | I, Me, You...etc. |
| Noun | N | All Nouns |
| Domain Suffix | DS | .com, .org, .us...etc. |
| Prefixes | DP | http, www. |
| Preposition | PP | For, of, about...etc. |
| How | QW_How | How, How far, How many, How much, How often |
| What | QW_What | What |
| When | QW_When | When |
| Where | QW_Where | Where |
| Who | QW_Who | Who |
| Which | QW_Which | Which |
| Question Words | QW | All question words |

4.1 Constructing Search Type Patterns and Term Category Taxonomy

In order to construct Search Type Patterns and term categories we have used 80,000 randomly selected queries from AOL 2006 datasets. We have taken the following steps:

Step 1- parsing the 80,000 queries and automatically extracting terms in the queries.

Step 2- manually performing initial categorization for the terms.

Step 3- processing the queries and converting each query to a Query Pattern. A Query Pattern (QP) is a representation of the original query where each term is replaced by a term category from the categories that we have constructed. $QP = \langle tc_1, tc_2, \dots, tc_n \rangle$. For example, the query: "Free Wallpapers" is converted to the Query Pattern: $\langle Adj_F + CN_OF \rangle$. A Query Pattern is an intermediate step towards reaching the final refined Search Type Patterns.

Step 4- grouping similar queries according to their Query Pattern. This reduced the size of the initial dataset significantly. For example, the two queries: "Who is Stephen Hawking" and "Who is Michael Phelps" both have the same Query Pattern: $\langle QW_Who + LV + PN_C \rangle$. The resulting set of Query Patterns is a much smaller representation of the original dataset.

Step 5- manually classifying each Query Pattern into one of the search types discussed in section 3.1. According to the semantics of the search types. For example, the Query Pattern: $\langle QW_Who + LV + PN_C \rangle$ is classified as *Informational-Directed-Closed*.

Step 6- To reduce the number of resulting patterns and to make them more generalized we performed a final step where we analysed Query Patterns in each search type separately and merged patterns that could be merged. Two Query Patterns $QP_x = \langle tc_{x1}, tc_{x2}, \dots, tc_{xn} \rangle$ and $QP_y = \langle tc_{y1}, tc_{y2}, \dots, tc_{yn} \rangle$ could be merged if for each $tc_{xi} \in QP_x$ and $tc_{yi} \in QP_y$ $tc_{xi} = tc_{yi}$ or there is a common super-category

tc_{sup} for both tc_{xi} and tc_{yi} . Such super-category might already exist or it might be created, in this case we merge the two Query Patterns QP_x and QP_y in one new pattern that contains tc_{sup} instead of tc_{xi} and tc_{yi} . For example, the four Query Patterns: $\langle CN_L + PP + PN_PB \rangle$ (representing the query: "Location of Eiffel Tower"), $\langle CN_L + PP + PN_G \rangle$ (representing the query: "Location of Kuwait"), $\langle CN_L + PP + PN_IOG \rangle$ (representing the query: "location of university of Florida"), and $\langle CN_L + PP + PN_CO \rangle$ (representing the query: "location of IBM") are merged into the Query Pattern: $\langle CN_L + PP + PN_CGIP \rangle$, since PN_PB , PN_G , PN_IOG and PN_CO term categories have the same super-category PN_CGIP . Note that this step has resulted in the final refined taxonomy of term categories presented in Figure 1 and Table 1.

The final set of Query Patterns after merging is called the Search Type Patterns. Note that if sub-categories that are being merged are not representing all the terms in the super-category we still use the super-category if we found that the new Query Pattern is valid for the Search Type. This helped in making our patterns covering more queries than those just being encountered in the input dataset.

As a result of applying the steps above we generated a database that contains all terms extracted from the dataset that we have used. We enriched this database by adding all possible terms in all the 7 main super-categories except the Proper Noun Category, since Proper Nouns are infinite. Note that although our solution does not require knowing all Proper Nouns, it is still capable of classifying queries that contain unrecognized Proper Nouns, as

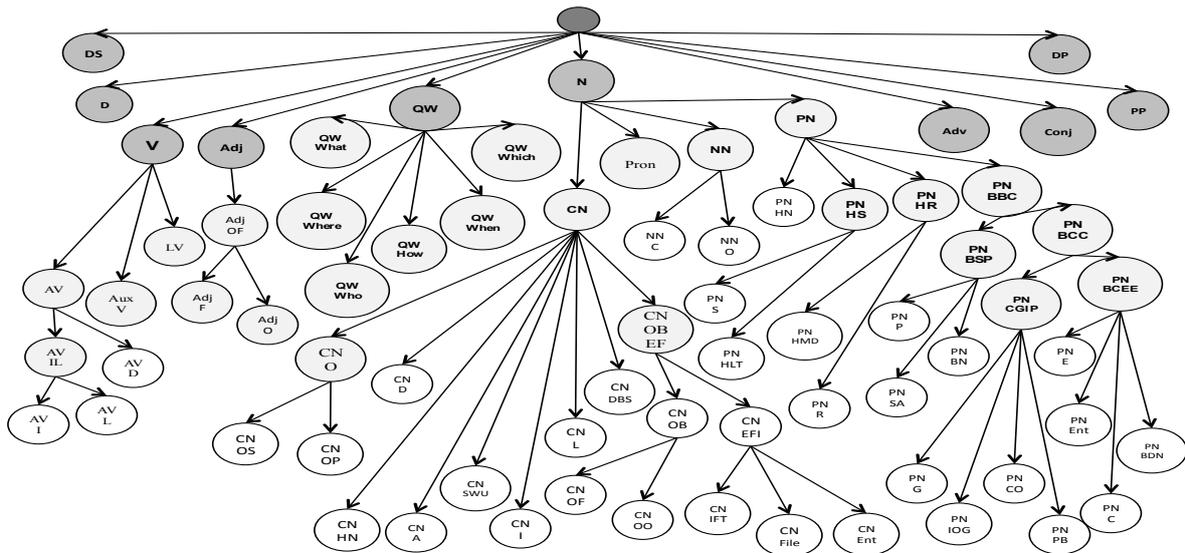


Figure 1: Terms Categorization.

we are going to illustrate in the next Section. The resulting database contains 10,440 terms classified into the classes shown in Table 1.

In addition to the term categories, we were able to identify 1182 Search Type Patterns. Table 1 and Table 2 show the distribution of these patterns by search type.

We validated our Search Type Patterns using a dataset containing 1953 queries from AOL that were manually classified and used in (Mendoza, et al., 2009).

Table 2: level 1 Search Type Patterns Distribution.

| Type of search | Total |
|----------------|-------|
| Informational | 838 |
| Transactional | 336 |
| Navigational | 8 |

Table 3: Level 2 and Level 3 Search Type Patterns Distribution.

| Type of search | Total |
|-----------------------------------|-------|
| Informational -List | 155 |
| Informational -Find | 164 |
| Informational -Advice | 121 |
| Informational -Undirected | 51 |
| Informational -Directed -Open | 113 |
| Informational -Directed -Closed | 234 |
| Transactional -Obtain -Online | 59 |
| Transactional -Obtain -Offline | 76 |
| Transactional -Interact | 28 |
| Transactional -Download -Free | 104 |
| Transactional -Download -not Free | 69 |

4.2 Classifying Search Engine Queries

Our solution automatically identifies and classifies user's queries by utilizing the Search Type Patterns and the term categories taxonomy presented in Figure 1. The proposed solution has three phases as shown in Figure 2:

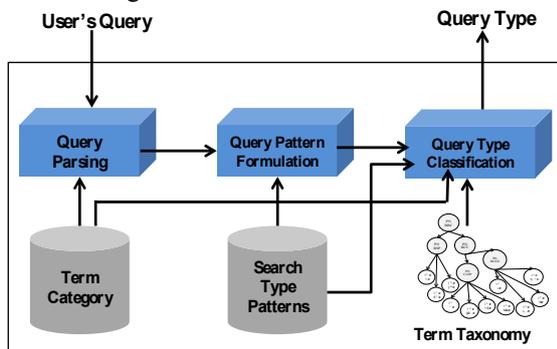


Figure 2: Proposed System Framework.

Phase 1- query parsing: this step is mainly responsible for extracting user's query terms. Unlike most other ir solutions, our solution does not destroy the query structure by removing stop-words and wh-question words. Such important query components are exploited in determining the query type. The system simply takes the user's query and parses it to facilitate the mapping of each word to the right category. For example given the two queries: query 1: "what is the capital of romania?" And query 2: "list of movies by steven spielberg" as inputs, the system extracts the following terms from query 1: "what", "is", "the", "capital", "of", "romania", and extracts the following terms from query 2: "list", "of", "movies", "by", "steven spielberg".

Phase 2- Query Pattern Formulation: the system converts the query to a Query Pattern by mapping terms in the query to corresponding term categories. First the system checks for compound terms (phrases) and then it processes single terms. The system maps each term to the most specific sub-category. If a term is not found in the terms database, the system assumes that the term is a Proper Noun, since Proper Nouns are infinite and we do not maintain an exhaustive list of them. After determining term category for all terms in the user query we then process consecutive terms that were identified as Proper Nouns. We convert such sequence of Proper Nouns to a single Proper Noun since no Search Type Pattern contains consecutive independent Proper Nouns.

The result of applying step 2 to query 1 is: "What"→QW_What, "is"→LV, "the"→D, "capital"→CN_OS, "of"→PP, "Romania"→PN_G. As a result, the Query Pattern for query 1 is: <QW_What + LV + D + CN_OS + PP + PN_G>. For query 2, if the terms database contains "Steven Spielberg", the system will be able to identify "Steven Spielberg" as a phrase and to determine its type as PN_C, hence the system will generate this Query pattern for query 2: <CN_I FT + PP + CN_Ent + PP + PN_C>. If "Steven Spielberg" was not contained in the terms database, the system assigned "Steven"→PN and "Spielberg"→PN, since both were not identified as any other type. The system then constructs this initial Query Pattern for query 2: <CN_I FT + PP + CN_Ent + PP + PN + PN> then it is modified to <CN_I FT + PP + CN_Ent + PP + PN> by merging the two consecutive Proper Nouns into a single Proper Noun.

Phase 3- Query Type Classification: In this step the system attempts to match the Query Pattern generated in step 2 with the most appropriate Search

Type Patterns to determine the Query type. For some Query Patterns, like the Query Pattern of query 1, this will be straightforward. This Query Pattern matches a Search Type Pattern in the Search Type *Informational-Directed-Closed*.

For other queries, like query 2, the Query Type does not fully match any Search Type Pattern. In this case we retrieve all Search Type Patterns that partially match the Query Pattern and we use the term categories taxonomy to determine which Search Type Patterns better match the Query Pattern. For example the Query Pattern $\langle CN_IFT + PP + CN_Ent + PP + PN \rangle$ of query 2 partially matches the Search Type Pattern $\langle CN_IFT + PP + CN_Ent + PP + PN_C \rangle$ from the *Informational-List* search type. And since PN_C is a sub-category of PN , the system classifies query 2 as *Informational-List*. Note that if the Query Pattern partially maps to a single search type, we can use this as a knowledge-learning step as the system might automatically add the new ambiguous term to the term categories database. This enriches the database of the system and reduces the cases of term ambiguity and partial query type matching in the future. If the Query Pattern partially maps to multiple search types, the system classify the query to more than one search types. This is a better treatment than considering the query totally vague and discarding it, as done by other solutions. This could be used to reduce the size of search engine result as we can provide the user with a very limited number of options that would reflect his/her intention.

5 EXPERIMENTS

We developed a prototype in Java to test our proposed solution. Our prototype utilizes the 1182 different Search Type Patterns that we have constructed and also use the taxonomy of term categories shown in Figure 1 and Table 1. This taxonomy of term categories contains 10,440 different terms and types.

To test the accuracy of our solution, 10,000 queries were randomly selected from AOL 2006 dataset and tested using the system. The selected queries are different from those used in constructing the Search Query Patterns. Results of the experiment show that our solution had identified and classified 7754 of the queries. After examining the remaining unclassified 2246 queries, we found that 927 of them were not identified due to vagueness or mistakes. This make the accuracy of the classification 85.5% of the queries without mistakes.

Table 4, shows classification detail by search type. Informational queries have the highest frequency with 4245 queries then transactional queries with 2783 queries. Navigational queries have the lowest frequency with only 726 queries. Table 5 shows the breakdown of the result to sub-categories.

Our experiments show that 944 out of the 1182 different Search Type Patterns were used in classifying the 10,000 queries that were used in our experiment.

Table 4: Query Classification Results.

| Type of search | Total |
|----------------|-------|
| Informational | 4245 |
| Transactional | 2783 |
| Navigational | 726 |

Table 5: Extended Classification Results.

| Type of search | Total |
|-----------------------------------|-------|
| Informational -List | 1117 |
| Informational -Find | 875 |
| Informational -Advice | 351 |
| Informational -Undirected | 986 |
| Informational -Directed -Open | 283 |
| Informational -Directed -Closed | 633 |
| Transactional -Obtain -Online | 860 |
| Transactional -Obtain -Offline | 726 |
| Transactional -Interact | 94 |
| Transactional -Download -Free | 548 |
| Transactional -Download -not free | 555 |

6 CONCLUSIONS

In this research, we have introduced a framework to automatically identify and classify search engine user queries. Unlike other solutions, our solution relies on both query terms and query structure in order to determine the user intent. We have categorized search queries through introducing Search Type Patterns. Our framework consists of three main steps: (1) parsing user's query, (2) formulating Query Patterns, and (3) Classifying query type.

Experiments show that our solution has achieved high accuracy in classifying queries. As a future work we will examine and analyze more queries from different search engine datasets in order to extend the ability of our system to identify more queries. We also plan to conduct more experiments on larger datasets and compare our results to results obtained from other approaches.

REFERENCES

- Ashkan, A., Clarke, C. L., Agichtein, E., & Guo, Q., 2009. Classifying and characterizing query intent. In *Advances in Information Retrieval* (pp. 578-586). Springer Berlin Heidelberg.
- Broder, A., 2002. A taxonomy of web search. In *ACM Sigir forum* (Vol. 36, No. 2, pp. 3-10). ACM.
- Bhatia, S., Brunk, C., & Mitra, P., 2012. Analysis and automatic classification of web search queries for diversification requirements. *Proceedings of the American Society for Information Science and Technology*, 49(1), 1-10.
- Baeza-Yates, R., Calderón-Benavides, L., & González-Caro, C., 2006. The intention behind web queries. In *String processing and information retrieval* (pp. 98-109). Springer Berlin Heidelberg.
- Beitzel, S. M., Jensen, E. C., Frieder, O., Grossman, D., Lewis, D. D., Chowdhury, A., & Kolcz, A., 2005. Automatic web query classification using labeled and unlabeled training data. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 581-582). ACM.
- Choo, C. W., Detlor, B., & Turnbull, D., 2000. Information seeking on the Web: An integrated model of browsing and searching. *firstmonday*, 5(2).
- Calderón-Benavides, L., Gonzalez-Caro, C., & Baeza-Yates, R., 2010. Towards a deeper understanding of the user's query intent. In *SIGIR 2010 Workshop on Query Representation and Understanding* (pp. 21-24).
- Hernández, D. I., Gupta, P., Rosso, P., & Rocha, M. A., 2012. Simple Model for Classifying Web Queries by User Intent.
- Jansen, B. J., & Booth, D., 2010. Classifying web queries by topic and user intent. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems* (pp. 4285-4290). ACM.
- Jansen, B. J., Booth, D. L., & Spink, A., 2008. Determining the informational, navigational, and transactional intent of Web queries. *Information Processing & Management*, 44(3), 1251-1266.
- Kathuria, A., Jansen, B. J., Hafernik, C., & Spink, A., 2010. Classifying the user intent of web queries using k-means clustering. In *Internet Research*, 20(5), 563-581.
- Kellar, M., Watters, C., & Shepherd, M., 2006. A Goal-based Classification of Web Information Tasks. *Proceedings of the American Society for Information Science and Technology*, 43(1), 1-22.
- Liu, Y., Zhang, M., Ru, L., & Ma, S., 2006. Automatic query type identification based on clickthrough information. In *Information Retrieval Technology* (pp. 593-600). Springer Berlin Heidelberg.
- Lee, U., Liu, Z., & Cho, J., 2005. Automatic identification of user goals in web search. In *Proceedings of the 14th international conference on World Wide Web* (pp. 391-400). ACM.
- Lewandowski, D., 2006. Query types and search topics of German Web search engine users. *Information Services and Use*, 26(4), 261-269.
- Lewandowski, D., Drechsler, J., & Mach, S., 2012. Deriving query intents from web search engine queries. *Journal of the American Society for Information Science and Technology*, 63(9), 1773-1788.
- Mendoza, M., & Zamora, J., 2009. Identifying the intent of a user query using support vector machines. In *String Processing and Information Retrieval* (pp. 131-142). Springer Berlin Heidelberg.
- Morrison, J. B., Pirolli, P., & Card, S. K., 2001, March. A taxonomic analysis of what World Wide Web activities significantly impact people's decisions and actions. In *CHI'01 extended abstracts on Human factors in computing systems* (pp. 163-164). ACM.
- Rose, D. E., Levinson, D., 2004. Understanding user goals in web search. In *Proceedings of the 13th international conference on World Wide Web* (pp. 13-19). ACM.
- Wu, D., Zhang, Y., Zhao, S., & Liu, T., 2010. Identification of Web Query Intent Based on Query Text and Web Knowledge. In *Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on* (pp. 128-131). IEEE.

A Domain Independent Double Layered Approach to Keyphrase Generation

Dario De Nart and Carlo Tasso

Artificial Intelligence Lab, Department of Mathematics and Computer Science, University of Udine, Udine, Italy
{dario.denart, carlo.tasso}@uniud.it

Keywords: Keyphrase Extraction, Keyphrase Inference, Information Extraction, Text Classification, Text Summarization.

Abstract: The annotation of documents and web pages with semantic metadata is an activity that can greatly increase the accuracy of Information Retrieval and Personalization systems, but the growing amount of text data available is too large for an extensive manual process. On the other hand, automatic keyphrase generation, a complex task involving Natural Language Processing and Knowledge Engineering, can significantly support this activity. Several different strategies have been proposed over the years, but most of them require extensive training data, which are not always available, suffer high ambiguity and differences in writing style, are highly domain-specific, and often rely on a well-structured knowledge that is very hard to acquire and encode. In order to overcome these limitations, we propose in this paper an innovative domain-independent approach that consists of an unsupervised keyphrase extraction phase and a subsequent keyphrase inference phase based on loosely structured, collaborative knowledge such as Wikipedia, Wordnik, and Urban Dictionary. This double layered approach allows us to generate keyphrases that both describe and classify the text.

1 INTRODUCTION

The tremendous and constant growth of the amount of text data available on the web has lead, in the last years, to an increasing demand for automatic summarization and information filtering systems. Such systems, in order to be effective and efficient, need metadata capable of representing text contents in a compact, yet detailed way.

As broadly discussed in literature and proven by web usage analysis (Silverstein et al., 1999), is particularly convenient for such metadata to come in the form of *KeyPhrases*(KP), since they can be very expressive (much more than single keywords), straightforward in their meaning, and have a high cognitive plausibility, because humans tend to think in terms of KPs rather than single keywords. In the rest of this paper we will refer to *KP generation* as the process of associating a meaningful set of KPs to a given text, regardless to their origin, while we will call *KP extraction* the act of selecting a set of KP from the text and *KP inference* the act of associating to the text a set of KP that may not be found inside it. KP generation is a trivial and intuitive task for humans, since anyone can tell at least the main topics of a given text, or decide whether it belongs to a certain domain (news item, scientific literature, narrative, etc., ...) or not,

but it can be extremely hard for a machine since most of the documents available lack any kind of semantic hint.

Over the years several authors addressed this issue proposing different approaches towards both KP extraction and inference, but, in our opinion, each one of them has severe practical limitations that prevent massive employment of automatic KP generation in *Information Retrieval*, *Social Tagging*, and *Adaptive Personalization*. Such limitations are the need of training data, the impossibility of associating to a given text keyphrases which are not already included in that text, the high domain specificity, and the need of structured, detailed, and extensive domain knowledge coded in the form of a thesaurus or an ontology. We claim that, in order to match the KP generation performances of a human expert, automatic KP generation systems should both extract and infer KPs, moreover such systems should be unsupervised and domain independent in order to be extensively used, since training data and domain ontologies are hard to obtain.

In order to support our claim we propose here an unsupervised KP generation method that consists of two layers of analysis: a KP Extraction phase and a KP inference one, based on Ontology Reasoning upon knowledge sources that though not being for-

mal ontologies can be seen as loosely structured ones. The first phase provides KPs extracted from the text, describing its content in detail, while the second provides more general KPs, chosen from a controlled dictionary, categorizing the text, rather than describing it.

The rest of the paper is organized as follows: in Section 2 we briefly introduce some related works; in Section 3 we present our keyphrase extraction technique; in Section 4 we illustrate our keyphrase inference technique; in Section 5 we discuss some experimental results and, finally, in Section 6 we conclude the paper.

2 RELATED WORK

Many works over the past few years have discussed different solutions for the problem of automatically tagging documents and Web pages as well as the possible applications of such technologies in the fields of Personalization and Information Retrieval in order to significantly reduce information overload and increase accuracy. Both keyphrase extraction and inference have been widely discussed in literature. Several different keyphrase extraction techniques have been proposed, which usually are structured into two phases:

- a *candidate phrase identification* phase, in which all the possible phrases are detected in the text;
- a *selection* phase in which only the most significant of the above phrases are chosen as keyphrases.

The wide span of proposed methods can be roughly divided into two distinct categories:

- *Supervised approaches*: the underlying idea of these methods is that KP Extraction can be seen as a *classification* problem and therefore solved with a sufficient amount of training data (manually annotated) and machine learning algorithms (Turney, 2000). Several authors addressed the problem in this direction (Turney, 1999) and many systems that implement supervised approaches are available, such as KEA (Witten et al., 1999), Extractor², and LAKE (DAvanzo et al., 2004). All the above systems can be extremely effective and, as far as reliable data sets are available, can be flawlessly applied to any given domain (Marujo et al., 2013), however requiring training data in order to work properly, implies two major drawbacks: (i) the quality of the extraction process relies on the quality of training data and (ii) a model trained on a specific domain just won't fit another application domain unless is trained again.

- *Unsupervised approaches*: this second class of methods eliminates the need for training data by selecting candidate KP according to some ranking strategy. Most of the proposed systems rely on the identification of *noun phrases*, i.e. phrases made of just nouns and then proceed with a further selection based on heuristics such as frequency of the phrase (Barker and Cornacchia, 2000) or upon phrase clustering (Bracewell et al., 2005). A third approach proposed by (Mihalcea and Tarau, 2004) and (Litvak and Last, 2008), exploits a graph-based ranking model algorithm, bearing much similarity to the notorious Page Rank algorithm, in order to select significant KPs and identify related terms that can be summarized by a single phrase. All the above techniques share the same advantage over the supervised strategies, that is being truly domain independent, since they rely on general principles and heuristics and therefore there is no need for training data.

Hybrid approaches have been proposed as well, incorporating semi-supervised domain knowledge in an otherwise unsupervised extraction strategy (Sarkar, 2013), but still remain highly domain-specific.

Keyphrase extraction, however, is severely limited by the fact it can ultimately return only words contained in the input document, which are highly prone to ambiguity and subject to the nuances of different writing styles (e.g: an author can write "mining frequent patterns" where another one would write "frequent pattern mining"). Keyphrase inference can overcome these limitations and has been widely explored in literature as well, spanning from systems that simply combine words appearing in the text in order to construct rather than extract phrases (Danilevsky et al., 2013) to systems that assign KPs that may built with terms that never appear in the document. In the latter case, KPs come from a controlled dictionary, possibly an ontology; in such case, a classifier is trained in order to find which entries of the exploited dictionary may fit the text (Dumais et al., 1998). If the dictionary of possible KPs is an ontology, its structure can be exploited in order to provide additional evidence for inference (Pouliquen et al., 2006) and, by means of ontological reasoning, evaluate relatedness between terms (Medelyan and Witten, 2006). In (Pudota et al., 2010) is discussed a KP inference technique based on a very specific domain ontology, written in the OWL language, in the context of a vast framework for personalized document annotation that combines both KP Extraction and inference. KP inference based on dictionaries, however, is strongly limited by the size, the domain coverage, and the specificity level of the considered dictionary.

3 SYSTEM OVERVIEW

In order to test our approach and to support our claims we developed a new version of the system presented in (Pudota et al., 2010). We introduce a new double-layered architecture and an original innovation, i.e. the exploitation of a number of generalist online External Knowledge Sources, rather than a formal domain specific ontology, in order to improve extraction quality, to infer meaningful KPs not included in the input text and to preserve domain independence.

In Figure 1 the overall organization of the proposed system is presented. It is constituted by the following main components:

- A *KP Extraction Module (KPEM)*, devoted to analyse the text and extract from it meaningful KPs. It is supported by some linguistic resources, such as a *POS tagger* (for the English Language) and a *Stopwords Database* and it accesses some online *External Knowledge Sources (EKSSs)* mainly exploited in order to provide support to the candidate KPs identified in the text (as explained in the following section). The KPEM receives in input an unstructured text and it produces in output a ranked list of KPs, which is stored in an *Extracted Keyphrases Data Base (EKPDDB)*.
- A *KP Inference Module (KPIM)*, which works on the KP list produced by the KPEM and it is devoted to infer new KPs, not already included in the input text. It relies on some ontological reasoning based on the access to the External Knowledge Sources, exploited in order to identify new concepts which are related to the ones referred to by the KPs previously extracted by the KPEM. Inferred KPs are stored in the *Inferred KP Data Base (IKPDDB)*.

The access to the online External Knowledge Sources is provided by a *Generalized Knowledge Gateway (GKG)*. The system is organized in the form of a Web service, allowing easy access to the KP Generation service to all kinds of clients.

The workflow of the system is intended as a simulation of the typical cognitive process that happens when we are asked to summarize or classify a text. At the beginning all of the text is read, then the KPEM identifies and ranks concepts included in the text, finally, the KPIM preprocesses the identified concepts in order to infer from them other concepts that may be tightly related or implied. The result of the process is a set of KPs that appear or do not appear in the text, thus mixing explicit and tacit knowledge.

4 PHRASE EXTRACTION

KPEM is an enhanced version of *DIKPE*, the unsupervised, domain independent KP extraction approach described in (Pudota et al., 2010) and (Ferrara and Tasso, 2013). In a nutshell, *DIKPE* generates a large set of candidate KPs; the exploited approach then merges different types of knowledge in order to identify meaningful concepts in a text, also trying to model a human-like KP assignment process. In particular we use: *Linguistic Knowledge* (POS tagging, sentence structure, punctuation); *Statistical Knowledge* (frequency, tf/idf,...); knowledge about the *structure* of a document (position of the candidate KP in the text, title, subtitles, ...); *Meta-knowledge* provided by the author (html tags,...); knowledge coming from *online external knowledge sources*, useful for validating candidate keyphrases which have been socially recognized, for example, in collaborative wikis (e.g. Wikipedia, Wordnik, and other online resources).

By means of the above knowledge sources, each candidate phrase, is characterized by a set of features, such as, for example:

- *Frequency*: the frequency of the phrase in the text;
- *Phrase Depth*: at which point of the text the phrase occurs for the first time: the sooner it appears, the higher the value;
- *Phrase Last Occurrence*: at which point of the text the phrase occurs for the last time: the later it appears, the higher the value;
- *Life Span*: the fraction of text between the first and the last occurrence of the phrase;
- *POS Value*: a parameter taking into account the grammatical composition of the phrase, excluding some patterns and assigning higher priority to other patterns (typically, for example but not exclusively, it can be relevant to consider the number of nouns in the phrase over the number of words in the phrase).
- *WikiFlag*: a parameter taking into account the fact that the phrase is or is not an entry of online collaborative external knowledge sources (EKSSs); the WikiFlag provides evidence of the social meaningfulness for a KP and therefore can be considered a feature based on general knowledge.

A weighted linear combination of the above features, called *Keyphraseness* is then computed and the KPs are sorted in descending keyphraseness order. The weight of each feature can be tuned in order to fit particular kinds of text, but, usually, a generalist preset can be used with good results. The topmost n KPs are finally suggested.

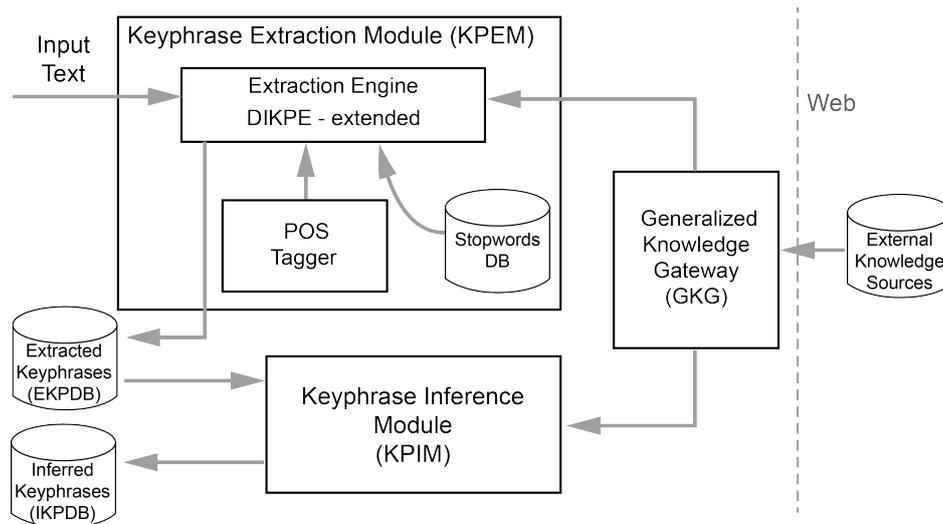


Figure 1: Architecture of the System.

In this work, we extended the DIKPE system with the GKG module, allowing access to multiple knowledge sources at the same time. We also added a more general version of the WikiFlag feature. This feature is computed as follows: if the phrase matches an entry in at least one of the considered external knowledge sources, then its value is set to 1, otherwise the phrase is split into its constituents and the WikiFlag value is set to the percentage corresponding to the number of terms that have a match in at least one of the considered external knowledge sources. By doing so, a KP that does not match as phrase, but is constituted by terms that match as single words, still gets a high score, but lower than a KP that features a perfect match. The WikiFlag feature is processed as all the other features, contributing to the computation of the keyphraseness and therefore influencing the ranking of the extracted KPs. The rationale of this choice is that a KP is important insofar it represents a meaningful concept or entity, rather than a random combination of words, and matching a whole phrase against a collaborative human-made knowledge source (as the EKSs are) guarantees that it makes better sense, providing a strong form of human/social validation. However, the WikiFlag does not prevent terms and phrases that are not validated by external knowledge to be suggested as KPs if they appear with significant frequency in significant parts of the document, which may be the case of newly introduced terminology or highly specific jargon. Exploiting the Wikiflag actually helps in reducing the tendency of the system to suggest typos, document parsing errors, random combinations of frequent non-stopwords terms, and other kinds of false positives.

Another improvement over the original DIKPE approach is represented by the fact that, instead of suggesting the top n KPs extracted, the new system evaluates the decreasing trend of Keyphraseness among ordered KPs, it detects the first significant downfall (detected by evaluation of the derivative function) in the keyphraseness value, and it suggests all the KPs occurring before that (dynamic) threshold. By doing so, the system suggests a variable number of high-scored KPs, while the previous version suggests a fixed number of KPs, that could have been either too small or too large for the given text.

5 PHRASE INFERENCE

The KP Inference Module (KPIIM), as well as the knowledge-based WikiFlag feature described in the previous section, rely on a set of external knowledge sources that are accessed via web. In the following we call *entity* any entry present in one or more EKSs; entities may have a complex structure, as well as include different kinds of data (e.g: text, pictures, videos, ...), however we are interested in the relationships occurring between entities rather than their content. EKSs may be online databases, such as Wordnet, linked data or traditional web resources as long as a dense link structure with some well-recognizable semantics is available. We assume that (i) there is a way to match extracted KPs with entities described in EKSs (e.g.: querying the exploited service using the KP as search key) and (ii) each one of the EKSs considered is organized according to some kind of hierarchy. Such hierarchy may be loose, but it must include some kind of *is-a* and *is-related* relationships, allowing us to infer,

for each entity, a set of parent and a set of related entities. Such sets may be void, since we do not assume each entity being necessarily linked to at least another one, nor the existence of a root entity that is ancestor of all the other entities in the ontology.

Even if such structure is loose, assuming its existence is nowadays not trivial at all; however, along with the growth of semantic web resources, an increasing number of collaborative resources allow users to classify and link together knowledge items, generating an increasing number of pseudo-ontologies. Clear examples of this trend are Wikipedia, where almost any article contains links to other articles and many articles are grouped into *categories*, and Wordnik, an online collaborative dictionary where any word is associated to a set of hypernyms, synonyms, hyponyms, and related terms. Recently also several entertainment sites, like Urban Dictionary, have begun to provide these possibilities, making them eligible knowledge sources for our approach. Knowledge sources may be either generalist (like Wikipedia), or specific (like the many domain-specific wikis hosted on *wikia.com*) and several different EKSs can be exploited at the same time in order to provide better results.

In the case of Wikipedia, parent entities are given by the *categories*, that are thematic groups of articles (i.e.: “Software Engineering” belongs to the “Engineering Disciplines” category). An entry may belong to several categories, for example the entry on “The Who” belongs to the “musical quartets” category as well as to the “English hard rock musical groups” one and the “Musical groups established in 1964” one. Related entities, instead, can be derived from links contained in the page associated to the given entity: such links can be very numerous and heterogeneous, but the most closely related ones are often grouped into one or more *templates*, that are the thematic collections of internal Wikipedia links usually displayed on the bottom of the page, as shown in Figure 2. For instance, in a page concerning a film director, it is very likely to find a template containing links to the all movies he directed or the actors he worked with.

Wordnik, instead, provides hierarchical information explicitly by associating to any entity lists of hypernyms (parent entities) and synonyms (related entities).

The inference algorithm considers the topmost half of the extracted KPs, that typically is still a significantly larger set than the one presented as output, and, for each KP that can be associated to an entity, retrieves from each EKS a set of parent entities and a set of related entities. If a KP corresponds to more than one entity on one or more EKSs, all of the retrieved

entities are taken into account. The sets associated to single KPs are then merged into a table of related entities and a table of parent entities for the whole text. Each retrieved entity is scored accordingly to the sum of the Keyphraseness value of the KPs from which it has been derived and then it is sorted by descending score. The top entries of such tables are suggested as meaningful KPs for the input document.

By doing so, we select only entities which are related to or parent of a significant number of hi-scored KPs, addressing the problem of polysemy among the extracted KP. For instance, suppose we extracted “Eiffel” and “Java Language” from the same text: they both are polysemic phrases since the first may refer to an ISO-standardized OO language as well as to a French civil engineer and architect and the latter to the programming language or to the language spoken in the island of Java. However, since they appear together, and they are both part of the “Object-Oriented Programming Languages” category in Wikipedia, it can be deduced that the text is about computer science rather than architecture or Indonesian languages.

6 EVALUATION

Formative tests were performed in order to test the accuracy of the inferred KPs and their ability to add meaningful information to the set of extracted KPs, regardless of the domain covered by the input text. Several data sets, dealing with different topics, were processed, article by article, with the same feature weights and exploiting Wikipedia and Wordnik as External Knowledge Source. For each article a list of extracted KPs and one of inferred KPs were generated, then the occurrences of each KP were counted, in order to evaluate which portion of the data set is covered by each KP. We call *set coverage* the fraction of the data set labelled with a single KP. Since the topics covered in the texts included in each data set are known a-priori, we expect the system to generate KPs that associate the majority of the texts in the data set to their specific domain topic.

The first data set contained 113 programming tutorials, spanning from brief introductions published on blogs and forums to extensive articles taken from books and journals, covering both practical and theoretical aspects of programming. A total of 776 KPs were extracted and 297 were inferred. In Table 1 are reported the most frequently extracted and inferred KPs. As expected, extracted KPs are highly specific and tend to characterize a few documents in the set (the most frequent KP covers just the 13% of the data set), while inferred ones provide an higher level

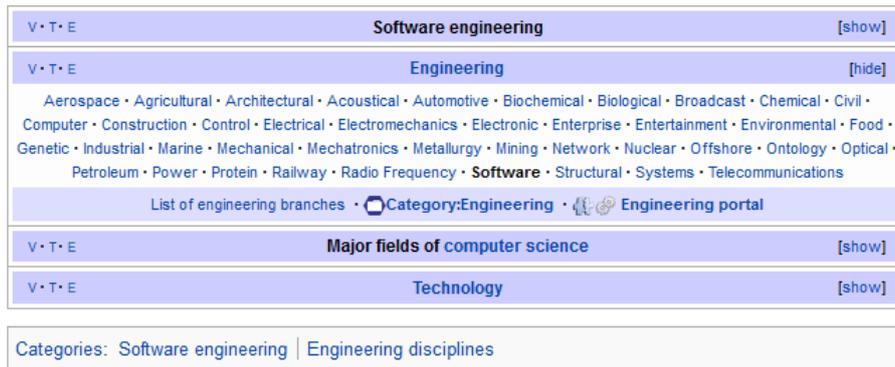


Figure 2: The lowest section of a Wikipedia page, containing templates (the “Engineering” template has been expanded) and categories (bottom line).

Table 1: The most frequently extracted and inferred KPs from the “programming tutorials” data set.

| Extracted Keyphrase | Set coverage | Inferred Keyphrase | Set Coverage |
|----------------------|--------------|-----------------------------|--------------|
| program | 0,13 | Mathematics | 0,47 |
| use | 0,12 | Programming language | 0,26 |
| function | 0,12 | move | 0,25 |
| type | 0,10 | Computer science | 0,22 |
| programming language | 0,10 | Set (mathematics) | 0,17 |
| programming | 0,08 | Data types | 0,15 |
| functions | 0,07 | Aristotle | 0,16 |
| class | 0,07 | Function (mathematics) | 0,14 |
| code | 0,06 | C (programming language) | 0,14 |
| COBOL | 0,06 | Botanical nomenclature | 0,12 |
| chapter | 0,05 | C++ | 0,11 |
| variables | 0,05 | Information | 0,08 |
| number | 0,05 | Java (programming language) | 0,08 |

Table 2: The most frequently extracted and inferred KPs from the “album reviews” data set.

| Extracted Keyphrase | Set coverage | Inferred Keyphrase | Set Coverage |
|---------------------|--------------|-------------------------|--------------|
| metal | 0,23 | Music genre | 1 |
| album | 0,21 | Record label | 0,97 |
| death metal | 0,17 | Record producer | 0,54 |
| black metal | 0,17 | United States | 0,48 |
| band | 0,16 | Studio album | 0,16 |
| bands | 0,08 | United Kingdom | 0,11 |
| death | 0,08 | Bass guitar | 0,09 |
| old school | 0,07 | Single (music) | 0,08 |
| sound | 0,06 | Internet Movie Database | 0,07 |
| albums | 0,05 | Heavy metal music | 0,07 |
| power metal | 0,05 | Allmusic | 0,06 |

of abstraction, resulting in an higher coverage over the considered data set. However some Inferred KPs are not accurate, such as “ Botanical nomenclature “ that clearly derive from the presence of terms such as “tree”, “branch”, “leaf”, and “forest” that are frequently used in Computer Science, and “Aristotele” which comes from the frequent references to Logic,

which Wikipedia frequently associates with the Greek philosopher.

Another data set contained reviews of 211 heavy metal albums published in 2013. Reviews were written by various authors, both professionals and non-professionals, and combine a wide spectrum of writing styles, from utterly specific, almost scientific, to

highly sarcastic, with many puns and popular culture references.

In Table 2 are reported the most frequently extracted and inferred KPs. All the documents in the set were associated with the Inferred KP “Music Genre” and the 97% of them with “Record Label”, which clearly associates the texts with the music domain. Evaluation and development are still ongoing and new knowledge sources, such as domain-specific wikis and Urban Dictionary, are being considered.

7 CONCLUSIONS AND FUTURE WORK

In this paper we proposed a truly domain independent approach to both KP extraction and inference, able to generate significant semantic metadata with two different layers of abstraction (phrase extraction and phrase inference) for any given text without need for training. The KP extraction part of the system provides a very fine level of detail, producing KPs that may not be found in a controlled dictionary (such as Wikipedia), but characterize the text. Such KPs are extremely valuable for the purpose of summarization and provide great accuracy when used as search keys. However, they are not widely shared, meaning, from an information retrieval point of view, a very poor recall. On the other hand, the KP inference part generates only KPs taken from a controlled dictionary (the union of the considered EKSs) that are more likely to be general, widely known and used, and, therefore, shared among a significant number of texts.

As shown in the previous section, our approach can annotate a set of documents with meaningful KPs, however, a few unrelated KPs may be inferred, mostly due to ambiguities of the text and to the generalist nature of the exploited online external knowledge sources. This unrelated terms, fortunately, tend to appear in a limited number of cases and to be clearly unrelated not only to the majority of the generated KPs, but also to each other. In fact, our next step in this research will be precisely to identify such false positives by means of an estimate of the *Semantic Relatedness* (Strube and Ponzetto, 2006), (Ferrara and Tasso, 2012) between terms in order to identify, for each generated KP, a list of related concepts and detect concept clusters in the document.

The proposed KP generation technique can be applied both in the Information Retrieval domain and in the Adaptive Personalization one. The previous version of the DIKPE system has already been integrated with good results in RES (De Nart et al., 2013), a personalized content-based recommender system for sci-

entific papers that suggests papers accordingly to their similarity with one or more documents marked as interesting by the user, and in the PIRATES framework (Pudota et al., 2010) for tag recommendation and automatic document annotation. We expect this extended version of the system to provide an even more accurate and complete KP generation and, therefore, to improve the performance of these existing systems, in this way supporting the creation of new Semantic Web Intelligence tools.

REFERENCES

- Barker, K. and Cornacchia, N. (2000). Using noun phrase heads to extract document keyphrases. In *Advances in Artificial Intelligence*, pages 40–52. Springer.
- Bracewell, D. B., Ren, F., and Kuriowa, S. (2005). Multilingual single document keyword extraction for information retrieval. In *Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on*, pages 517–522. IEEE.
- Danilevsky, M., Wang, C., Desai, N., Guo, J., and Han, J. (2013). Kert: Automatic extraction and ranking of topical keyphrases from content-representative document titles. *arXiv preprint arXiv:1306.0271*.
- DAvanzo, E., Magnini, B., and Vallin, A. (2004). Keyphrase extraction for summarization purposes: The lake system at duc-2004. In *Proceedings of the 2004 document understanding conference*.
- De Nart, D., Ferrara, F., and Tasso, C. (2013). Personalized access to scientific publications: from recommendation to explanation. In *User Modeling, Adaptation, and Personalization*, pages 296–301. Springer.
- Dumais, S., Platt, J., Heckerman, D., and Sahami, M. (1998). Inductive learning algorithms and representations for text categorization. In *Proceedings of the seventh international conference on Information and knowledge management*, pages 148–155. ACM.
- Ferrara, F. and Tasso, C. (2012). Integrating semantic relatedness in a collaborative filtering system. In *Mensch & Computer Workshopband*, pages 75–82.
- Ferrara, F. and Tasso, C. (2013). Extracting keyphrases from web pages. In *Digital Libraries and Archives*, pages 93–104. Springer.
- Litvak, M. and Last, M. (2008). Graph-based keyword extraction for single-document summarization. In *Proceedings of the workshop on multi-source multilingual information extraction and summarization*, pages 17–24. Association for Computational Linguistics.
- Marujo, L., Gershman, A., Carbonell, J., Frederking, R., and Neto, J. P. (2013). Supervised topical key phrase extraction of news stories using crowdsourcing, light filtering and co-reference normalization. *arXiv preprint arXiv:1306.4886*.
- Medelyan, O. and Witten, I. H. (2006). Thesaurus based automatic keyphrase indexing. In *Proceedings of the 6th*

- ACM/IEEE-CS joint conference on Digital libraries*, pages 296–297. ACM.
- Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. In *Proceedings of EMNLP*, volume 4. Barcelona, Spain.
- Pouliquen, B., Steinberger, R., and Ignat, C. (2006). Automatic annotation of multilingual text collections with a conceptual thesaurus. *arXiv preprint cs/0609059*.
- Pudota, N., Dattolo, A., Baruzzo, A., Ferrara, F., and Tasso, C. (2010). Automatic keyphrase extraction and ontology mining for content-based tag recommendation. *International Journal of Intelligent Systems*, 25(12):1158–1186.
- Sarkar, K. (2013). A hybrid approach to extract keyphrases from medical documents. *arXiv preprint arXiv:1303.1441*.
- Silverstein, C., Marais, H., Henzinger, M., and Moricz, M. (1999). Analysis of a very large web search engine query log. In *ACM SIGIR Forum*, volume 33, pages 6–12. ACM.
- Strube, M. and Ponzetto, S. P. (2006). Wikirelate! computing semantic relatedness using wikipedia. In *AAAI*, volume 6, pages 1419–1424.
- Turney, P. D. (1999). Learning to extract keyphrases from text. national research council. *Institute for Information Technology, Technical Report ERB-1057*.
- Turney, P. D. (2000). Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336.
- Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., and Nevill-Manning, C. G. (1999). Kea: Practical automatic keyphrase extraction. In *Proceedings of the fourth ACM conference on Digital libraries*, pages 254–255. ACM.

A Methodology to Measure the Semantic Similarity between Words based on the Formal Concept Analysis

Yewon Jeong, Yiyeon Yoon, Dongkyu Jeon, Youngsang Cho and Wooju Kim

Department of Information & Industrial Engineering, Yonsei University, Seoul, Korea
ywjeong88@gmail.com, pryiyeeon@nate.com, jdkclub85@gmail.com, {y.cho, wkim}@yonsei.ac.kr

Keywords: Query Expansion, Formal Concept Analysis, Semantic Similarity, Keyword-based Web Documents.

Abstract: Recently, web users feel difficult to find the desired information on the internet despite a lot of useful information since it takes more time and effort to find it. In order to solve this problem, the query expansion is considered as a new alternative. It is the process of reformulating a query to improve retrieval performance in information retrieval operations. Although there are a few techniques of query expansion, synonym identification is one of them. Therefore, this paper proposes the method to measure the semantic similarity between two words by using the keyword-based web documents. The formal concept analysis and our proposed expansion algorithm are used to estimate the similarity between two words. To evaluate the performance of our method, we conducted two experiments. As the results, the average of similarity between synonym pairs is much higher than random pairs. Also, our method shows the remarkable performance in comparison with other method. Therefore, the suggested method in this paper has the contribution to find the synonym among a lot of candidate words.

1 INTRODUCTION

Recently, the useful information on the internet has been increasing due to the rapid development of web. However, users feel difficult to find the desired information on the internet because it takes more time and efforts. In order to solve this problem, the query expansion is considered as a new alternative. It helps user to find the desired results and improve the effectiveness of retrieval. As the process of reformulating a query, the query expansion improves retrieval performance in information retrieval operations (Vechtomova and Wang, 2006). Thus, in the search engines, it involves evaluating a user's input and expanding the search query to match additional documents. Even if there are a few techniques of the query expansion, the synonym identification is one of them.

Finding synonym on the basis of subjective intuitions is considered as a daunting task. This is the reason of that it is hard to define the synonym due to a property that has no clear-cut boundaries (Baroni and Bisi, 2004). Therefore, this paper proposes the method to automatically measure how much two words have the semantically similar relation by using keyword-based web documents.

There are a lot of web documents which have tagged words like papers. Therefore, this paper applied the paper keywords to calculate the similarity between two words through the formal concept analysis (FCA).

The next section introduces the related work of the formal concept analysis and other similarity measurements. The section 3 provides a detailed explanation of methodology to measure similarity between two words. The section 4 presents the result of experiments to evaluate performance of our method. Finally, we draw the conclusion and suggest future work in the section 5.

2 RELATED WORKS

2.1 Formal Concept Analysis

The formal concept analysis is a mathematical approach which is used for conceptual data analysis (Ganter et al., 2005). It has been studied in diverse fields such as data mining, conceptual modelling, software engineering, social networking and the semantic web (Alqadah and Bhatnagar, 2011). It is good to analyse and manage structured data

(Wormuth and Becker, 2004). Thus, it helps user to structure an interest domain (Ganter et al., 1997, Wille, 2009). It models the world of data through the use of objects and attributes (Cole and Eklund, 1999). Ganter et al.(1999) applied the concept lattice from the formal concept analysis. This approach has an advantage that users can refine their query by searching well-structured graphs. These graphs, known as formal concept lattice, are composed of a set of documents and a set of terms. Effectively, it reduces the task of setting bound restrictions for managing the number of documents to be retrieved required (Tam, 2004).

2.2 Related Works of Similarity Measure between Two Words

Traditionally, a number of approaches to find synonym have been published. The methodology to automatically discover synonym from large corpora have been popular topic in a variety of language processing (Sánchez and Moreno, 2005, Senellart and Blondel, 2008, Blondel and Senellart, 2011, Van der Plas and Tiedemann, 2006). There are two kinds of approaches to identify synonyms.

The first kind of approaches uses a general dictionary (Wu and Zhou, 2003). In the area of synonym extraction, it is common to use lexical information in dictionary (Veronis and Ide, 1990). In dictionary-based case, a similarity is decided on definition of each word in a dictionary. This kind of approaches is conducted through learning algorithm based on information in the dictionary (Lu et al., 2010, Vickrey et al., 2010). Wu and Zhou (2003) proposed a method of synonym identification by using bilingual dictionary and corpus. The bilingual approach works on as follows: Firstly, the bilingual dictionary is used to translate the target word. Secondly, the authors used two bilingual corpora that mean precisely the same. And then, they calculated the probability of the coincidence degree. The result of the bilingual method is remarkable in comparison with the monolingual cases. Another research builds a graph of lexical information from a dictionary. The method to compute similarity for each word is limited to nearby words of graph. This similarity measurement was evaluated on a set of related terms (Ho and Fairon, 2004).

The second kind of approaches to identity synonym considers context of the target word and computes a similarity of lexical distributions from corpus (Lin, 1998). In the case of distributional approaches, a similarity is decided on context. Thus, it is important to compute how much similar words

are in a corpus. The approach of distributional similarity for synonym identification is used in order to find related words (Curran and Moens, 2002). There has been many works to measure similarity of words, such as distributional similarity (Lin et al., 2003). Landauer and Dumais (1997) proposed a similarity measurement to solve TOEFL tests of synonym by using latent semantic analysis (Landauer and Dumais, 1997). Lin (1998) proposed several methodologies to identify the most probable candidate among similar words by using a few distance measures. Turney (2001) presented PMI and IR method which is calculated by data from the web. He evaluated this measure on the TOEFL test in which the system has to select the most probable candidate of the synonym among 4 words. Lin et al. (2003) proposed two ways of finding synonym among distributional related words. The first way is looking over the overlap in translated texts of semantically similar words in multiple bilingual dictionaries. The second is to look through designed patterns so as to filter out antonyms.

There are a lot of researches for measuring similarity to identify the synonym. However, the use of dictionary has been applied to a specific task or domain(Turney, 2001). Hence, these existing researches are hard to be applied in the changeable web. And, the context-based similarity method deals with unstructured web documents and it takes much time to analysis since it needs to pre-treatment such as morphological analysis. Therefore, this paper proposes a methodology to automatically measure the semantically similar relation between two words by using keyword-based structured data from web.

3 METHOD TO MEASURE SIMILARITY

In this section, we demonstrate the method to measure semantic similarity between two distinct words. This paper defined the ‘query’ as the target word that we would like to compute the semantic similarity. A pair of queries is defined as $Q = (q_i, q_j)$ which is the set of two different words q_i and q_j .

The overall procedure to estimate semantic similarity between two queries of Q is composed of three phases as shown in the Figure 1; preprocessing, analysis and calculation phase. In the preprocessing phase, base data for the analysis are collected and refined on each query. Let us assume that the query pair is $Q=(contamination, pollution)$. The set of web

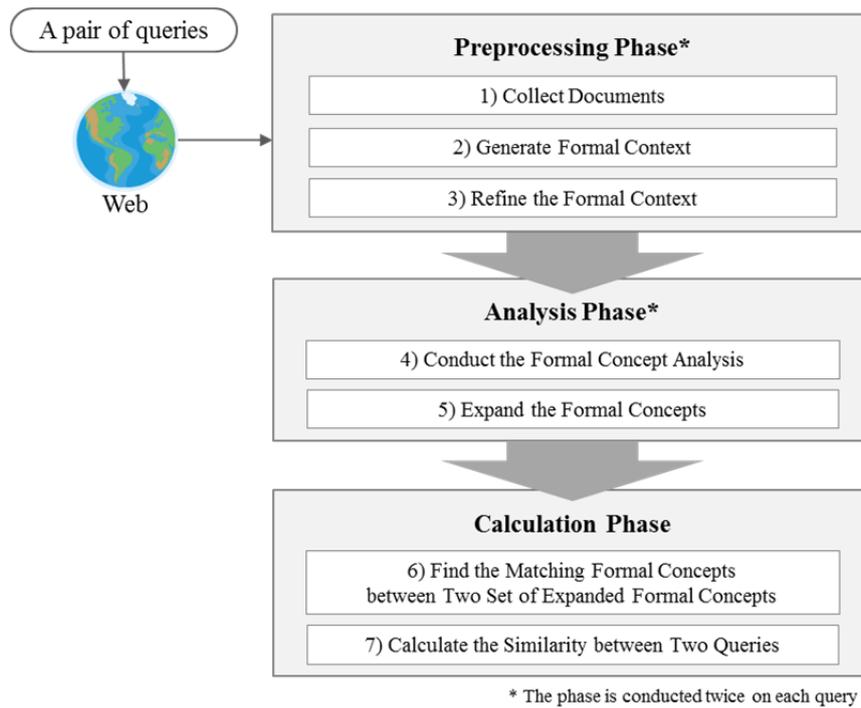


Figure 1: The overall procedure to calculate the semantic similarity between two queries.

documents for each queries *contamination* and *pollution* are collected respectively. The formal context for each query is constructed based on the set of collected web documents, tags and binary relations. Finally, the generated formal contexts are refined according to two rules which are introduced in the section 3.1.2. In the analysis phase, we apply FCA and expansion algorithm to each refined formal context. Implicit concepts from formal concept are derived through the expansion algorithm which helps us to compare queries in-depth. In the final phase, we calculate the semantic similarity of the pair of queries. On the basis of expanded formal concepts, we can examine how many concepts are duplicated by considering the matching concepts.

3.1 Preprocessing Phase

In order to measure the similarity between two queries, web documents which have the keywords should be collected on each query. And the keywords of collected documents should include the query. From these documents, we can get information about relation between documents and tagged words and also can make the formal context.

3.1.1 Generation of the Formal Context

A formal context is represented through a

two-dimensional matrix X . In general, the column and row of X indicate objects and attributes respectively. An object is a collected web document and an attribute is one of the tagged words. Table 1 shows the example of the formal context given the $Q = (contamination, pollution)$. The checkmarks in the table mean whether the object contains attributes or not. In the case of $q_i = contamination$, as shown in Table 1, the document d_1 has four attributes such as *contamination*, *insulators*, *solutions* and *flashover*. The each set of attributes and objects are defined as follows:

$$A^{q_i} = \{contamination, insulators, humidity, solutions, flashover, power lines\} \quad (1)$$

$$O^{q_i} = \{d_1, d_2, d_3, d_4, d_5\} \quad (2)$$

$$A^{q_j} = \{pollution, insulators, etching, solutions, falshover, iron\} \quad (3)$$

$$O^{q_j} = \{d_1, d_2, d_3, d_4, d_5\} \quad (4)$$

A^{q_i} is the set of attributes and O^{q_i} is the set of objects when q_i is given. A^{q_i} is composed of tags from the collected documents and O^{q_i} consists of the documents which is represented d_i .

Table 1: Examples of formal contexts.

| $q_i = \text{contamination}$ | | | | | | |
|------------------------------|--------------------|-----------------|---------------|----------------|----------------|----------------|
| | contami- nation | Insulat- ors | humidit- y | solution- s | flashov- er | power lines |
| d ₁ | √ | √ | | √ | √ | |
| d ₂ | √ | | √ | √ | | |
| d ₃ | √ | √ | | | | √ |
| d ₄ | √ | | √ | | √ | √ |
| d ₅ | √ | √ | √ | | | √ |

| $q_j = \text{pollution}$ | | | | | | |
|--------------------------|-----------|-----------------|---------|----------------|----------------|------|
| | pollution | Insulat- ors | etching | solution- s | flashov- er | iron |
| d ₁ | √ | | √ | | | √ |
| d ₂ | √ | | | √ | √ | |
| d ₃ | √ | √ | | | √ | |
| d ₄ | √ | | | √ | | √ |
| d ₅ | √ | √ | | √ | √ | |

3.1.2 Refinement of the Formal Context

After two formal contexts are generated, the refinement procedure is required for two reasons. Our research supposes that the more semantically similar relation two queries have, the more matching tagged words they have. This study ultimately wants to know how many words are matched between tagged words from two queries. Therefore, the attribute which is the same with query is unnecessary in this comparison procedure. The first refinement rule is to remove ‘query’ from attribute set A , and then, the second rule is to remove attributes which are contained in less than two documents. The reason is that these attributes have relatively weak effects to this method, and also it is helpful to save the process time and system cost by reducing the size of formal context. The summary of refinement procedure is as follows:

1. Removing the *query* from A (the set of attributes).
2. Removing the attributes contained in less than two web documents.

Table 2 is an example of refined context when the query is *contamination* and *pollution*. Because the *contamination* is given by q_i , the attribute *contamination* is removed by rule 1. For the same reason, the attribute *pollution* is also removed. Since the number of web documents contained in *etching* is less than 2, the attribute *etching* is removed by rule 2.

Table 2: Examples of refined formal contexts.

| $q_i = \text{contamination}$ | | | | | |
|------------------------------|------------|----------|-----------|-----------|----------------|
| | insulators | humidity | solutions | flashover | power lines |
| d ₁ | √ | | √ | √ | |
| d ₂ | | √ | √ | | |
| d ₃ | √ | | | | √ |
| d ₄ | | √ | | √ | √ |
| d ₅ | √ | √ | | | √ |

| $q_j = \text{pollution}$ | | | | |
|--------------------------|------------|-----------|-----------|------|
| | insulators | solutions | flashover | iron |
| d ₁ | | | | √ |
| d ₂ | | √ | √ | |
| d ₃ | √ | | √ | |
| d ₄ | | √ | | √ |
| d ₅ | √ | √ | √ | |

3.2 Analysis Phase

In this section, we introduce the analysis phase of this method. First, the formal concept analysis is conducted based on each formal context on q_i and q_j . However, a concept from the formal concept analysis has only a few implicit concepts. Thus, we expand the formal concepts through our proposed expansion algorithm.

3.2.1 Formal Concept Analysis

To measure the similarity between the two queries, formal concept analysis should be performed on each formal context of q_i and q_j . According to these analysis procedures, two sets of formal concepts are generated by using formal concept analysis (Ganter et al., 1997). When the query q_i is given, a set of formal concepts is generated by formal concept analysis as follows:

$$S(FC_k^{q_i}) = \{FC_1^{q_i}, FC_2^{q_i}, \dots, FC_n^{q_i}\} \quad (5)$$

where $k = 1, \dots, n$

In this equation, $S(FC_k^{q_i})$ is the set of formal concepts and $FC_k^{q_i}$ is the k th formal concept. And, n is the number of formal concepts from the formal context. A formal concept is composed of an intent and extent as demonstrated in (6):

$$FC_k^{q_i} = \{I_k^{q_i}, E_k^{q_i}\} \quad \text{where } k = 1, \dots, n \quad (6)$$

In this formula, $I_k^{q_i}$ is an intent of the $FC_k^{q_i}$ and $E_k^{q_i}$ is an extent. The intent is subset of the attribute set A^{q_i} which is the keyword set. And, extent is subset of

object set O^{q_i} which is the set of documents. Every object in $E_k^{q_i}$ has every attribute in $I_k^{q_i}$ by the property of formal concept analysis. Thus, $FC_k^{q_i}$ is a concept that implicates that the objects in $E_k^{q_i}$ have the common attributes in $I_k^{q_i}$.

From a set of formal concepts, we can get each set of intent on certain query. A set of $I_k^{q_i}$ is denoted as I^{q_i} :

$$I^{q_i} = \{I_1^{q_i}, I_2^{q_i}, \dots, I_n^{q_i}\} \quad (7)$$

where $I_k^{q_i} \subset P(A^{q_i})$

An element of I^{q_i} is subset of A^{q_i} and intent of each formal concepts. This set of intents is used when we calculate similarity between two set of formal concepts.

3.2.2 Expansion Algorithm

There are a few implicit concepts in a formal concept. Let us assume that a concept has the subsets of intent of other concepts. If it has the same extent each other, it is not generated by formal concept analysis. Therefore, we need to expand formal concept in order to compare them in depth. The detail procedure is as follows:

1. Find a formal concept (FC) which has the most size of intent from the set of formal concepts (FCS).
2. Get an extent (EXT) and intent (INT) from the FC .
3. Generate the subset of FC of which size is $n-1$ when the size of intent is n , and define it as $INTS$.
4. Confirm whether $INTS[i]$ (an element of $INTS$) is in the FCS .
5. If it isn't, add the expanded concept which has $INTS[i]$ and EXT .

6. Repeat this procedure until all of the formal concepts are expanded.

Firstly, the algorithm finds a formal concept which has the largest intent size. It is represented by the dotted outline in result of FCA in Figure 2. The intent size of this concept is 3, so generate subset of which size is 2. Then 3 subsets of an intent like $\{solutions, flashover\}$, $\{solutions, insulators\}$ and $\{flashover, insulators\}$ are made. Among these subsets, a subset $\{solutions, insulators\}$ doesn't exist in original set of formal concepts. Therefore, a new concept which consists of $\{solutions, insulators\}$, $\{d_5\}$ could be generated.

If the formal concepts go through expansion procedure, some concepts are generated. The Figure 3 shows examples of the expanded concepts lattice. The coloured boxes are the newly generated concepts. In this figure, (a) is a concept lattice of a context when the query q_i is contamination. There are 6 concepts made by expansion. And, (b) is a concept lattice of a context when the query q_j is pollution. Two concepts are generated. The expansion of formal concepts is helpful to compare them because implicit concepts can be found.

3.3 Calculation Phase

Suppose that there are the two queries denoted by q_i and q_j . The semantic similarity between q_i and q_j is calculated based on comparison of two sets of formal concepts. To compare them, we need to find the duplicated formal concepts.

3.3.1 Matching Formal Concepts

If there are two sets of formal concepts, the concepts which have the same intent are called to 'matching

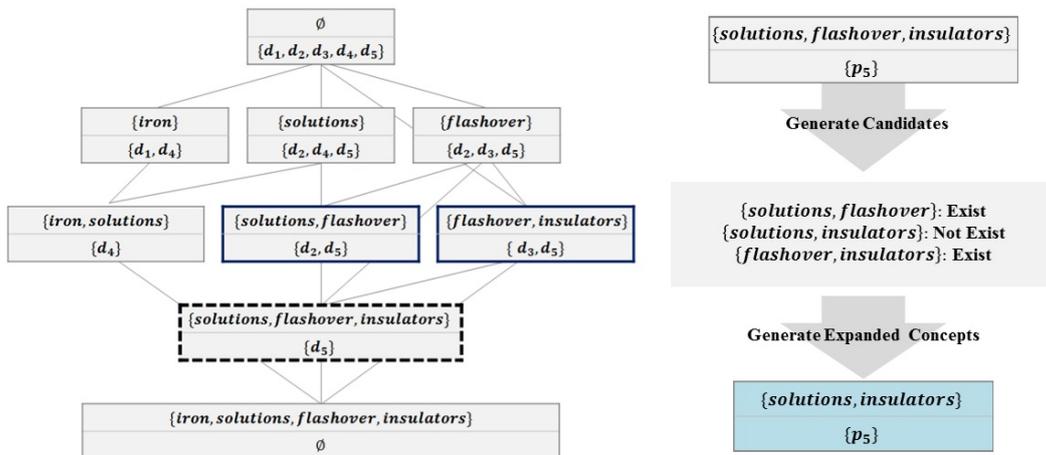


Figure 2: An example of expansion process.

concepts'. In other word, it means that concepts have the same intent from $FC_k^{q_i}$ and $FC_k^{q_j}$ respectively. In Figure 3, the concepts marked as bold outline are the matching concepts. When two queries, q_i and q_j , are given, the set of matching concepts is as follows:

$$S(MC_z^{q_{ij}}) = \{MC_1^{q_{ij}}, MC_2^{q_{ij}}, \dots, MC_c^{q_{ij}}\} \quad (8)$$

$S(MC_z^{q_{ij}})$ is a set of matching concepts and $MC_z^{q_{ij}}$ is the z th matching concept. And, c is the number of matching concepts. A matching concept is composed of an intent and two extents as follows:

$$MC_z^{q_{ij}} = \{I_z^*, E_z^{*q_i}, E_z^{*q_j}\} \quad (9)$$

I^* is the intersection of I^{q_i} and I^{q_j} . $E_z^{*q_i}$ is the extent when the intent is I_z^* and the q_i is given. Also, $E_z^{*q_j}$ is the extent when the intent is also I_z^* and the q_j is given. The function $MapFunc$ is a function to find an extent corresponding with a certain intent given query. The formulas are as follows:

$$I^* = (I^{q_i} \cap I^{q_j}) \quad (10)$$

$$E_z^{*q_i} = MapFunc(I_z^*, q_i) \quad (11)$$

3.3.2 Calculation of Semantic Similarity

If we gain the set of matching concepts, we can estimate the similarity between two queries, q_i and q_j . A measure of similarity is defined as:

$$Similarity(q_i, q_j) = \frac{\sum_{z=1}^c \left\{ \left(|I_z^*| \times |E_z^{*q_i}| \right) + \left(|I_z^*| \times |E_z^{*q_j}| \right) \right\}}{\sum_{x=1}^n \left(|I_x^{q_i}| \times |E_x^{q_i}| \right) + \sum_{y=1}^m \left(|I_y^{q_j}| \times |E_y^{q_j}| \right)} \times 100 \quad (12)$$

It is the measure to calculate how many concepts are duplicated. In this formula, we multiply the number of intent elements by the number of extent elements because the concepts that have the bigger size of an intent or extent have a great effect on measure. This similarity has range from zero to 100. If the all concepts are the same the similarity is 100. And if

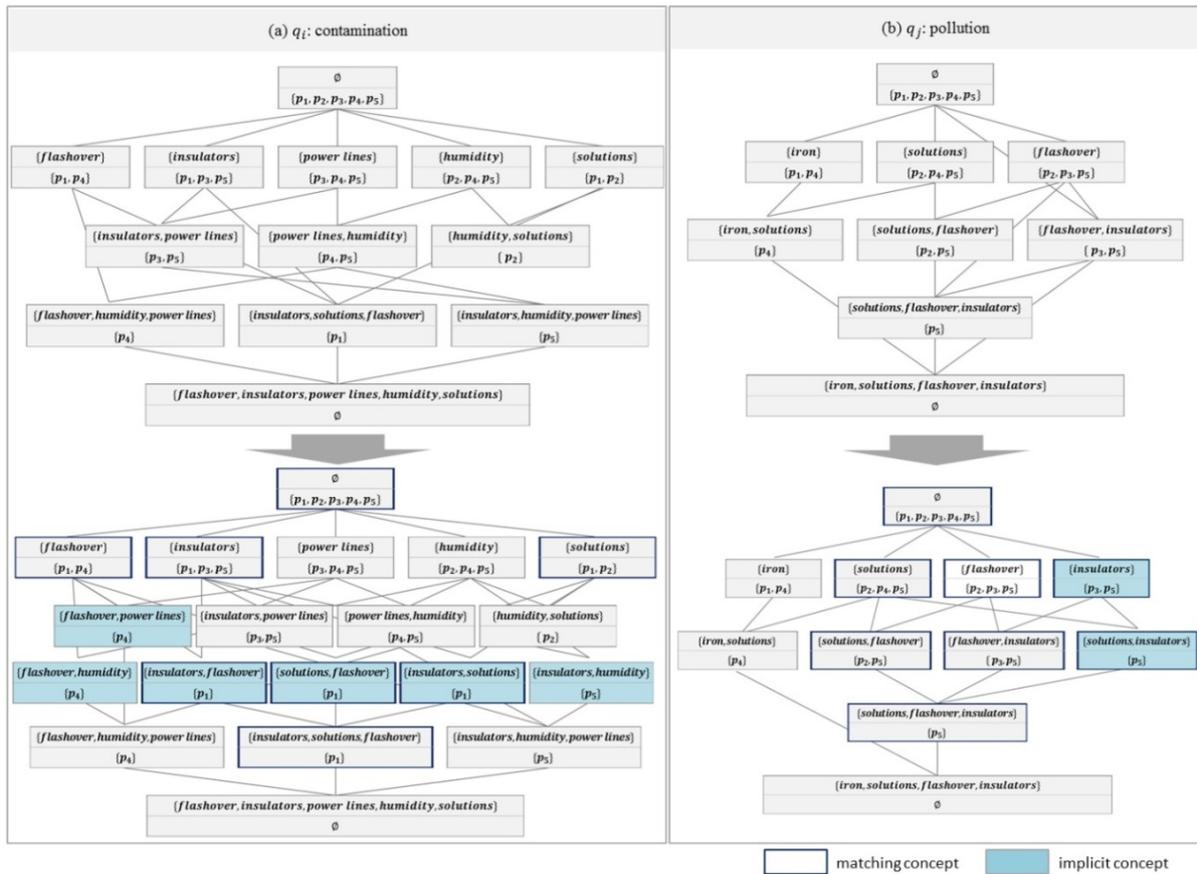


Figure 3: An example of expansion and matching concepts.

there are not duplicated concepts, the result would be zero.

4 EMPIRICAL EVALUATION

In order to evaluate the effectiveness of our method, we had the two performance evaluations. Firstly, we compared the similarity between two types of query pairs; one is the set of synonym pairs and the other is based on the randomly selected pairs. Secondly, we used the type of TOEFL synonym questions to verify the performance of this method.

4.1 Synonym Pairs Vs. Random Pairs

We prepared the 20 word pairs composed of 10 synonym pairs and 10 random pairs. In order to make formal contexts about queries, we collected papers tagged by each query from the IEEE Xplore website.

This paper shows the result of 10 experiments based on synonym pairs. The result of evaluation is shown in Table 3. The best resulted synonym pair scored as 5.22 is (*optimization, optimisation*). This pair has six matched formal concepts. We could know that it has the same meaning and significantly similar relation. The worst resulted synonym pair scored as 0.59 is (*validation, verification*) and has three matching formal concepts. This pair has weak similarity relation.

Table 3: The results of experiment (synonym pairs).

| No. | Synonym pairs | | Similarity (q_i, q_j) |
|---------|----------------|-----------------|------------------------------|
| | q_i | q_j | |
| 1 | partition | partitioning | 0.60 |
| 2 | optimization | optimisation | 5.22 |
| 3 | classification | categorization | 4.13 |
| 4 | cryptography | steganography | 1.71 |
| 5 | reliability | dependability | 1.17 |
| 6 | cluster | clustering | 4.95 |
| 7 | contamination | pollution | 0.87 |
| 8 | validation | verification | 0.59 |
| 9 | encoding | encryption | 1.45 |
| 10 | experiment | experimentation | 3.93 |
| Average | | | 2.46 |

In addition, we have experiment with 10 random pairs. The result is shown in Table 4. The average of all of the random pairs is approximately 0.37. The best resulted random pair scored as 0.99 is

(*normalization, segmentation*). It has six matching formal concepts. Although this query pair is not synonym, we can understand that they have a little relevant relation. There are the three worst results scored as zero and this pairs are composed of completely unrelated tags. (*integration, forecasting*), (*lifetime, authorization*) and (*correlation, evolution*) are unrelated pairs of experiment results. They don't have any common concepts each other and we could know that they don't have any semantic relations between them.

Table 4: The results of experiment (random pairs).

| No. | Random pairs | | Similarity (q_i, q_j) |
|---------|----------------|----------------|------------------------------|
| | q_i | q_j | |
| 1 | aggregation | android | 0.62 |
| 2 | calibration | internet | 0.25 |
| 3 | transportation | biometrics | 0.35 |
| 4 | context | innovation | 0.99 |
| 5 | integration | forecasting | 0.00 |
| 6 | lifetime | authorization | 0.00 |
| 7 | visualization | entropy | 0.61 |
| 8 | correlation | evolution | 0.00 |
| 9 | normalization | segmentation | 0.67 |
| 10 | sorting | authentication | 0.16 |
| Average | | | 0.37 |

While the average of similarity between synonym pairs is about 2.46, the average of random pairs is about 0.37. And it shows the remarkable difference between two types of pairs. Therefore, the method to measure similarity relation has the contribution to find the synonym among a lot of candidates.

4.2 TOEFL Synonym Test

We prepared the 9 TOEFL synonym questions to find the synonym of the target word. One question is composed of a target word and four candidate words. And, we measured the similarity between the target word and each candidate word. In order to compare the performance with the related works, we used the AVMI(Baroni and Bisi, 2004) and cosine similarity to compute similarity. In order to make contexts, we also collect papers from the IEEE Xplore website. And the result of experiments is shown as the Table 5. Our method has the 100 percentage of correct answers, but the AVMI and cosine similarity had the 78%, 89% performance respectively. It is a remarkable result in comparison with existing researches.

Table 5: Result of TOEFL Synonym Test.

| Target word | Candidate words | Our method | AVMI | Cosine similarity |
|----------------|------------------------|--------------|--------------|-------------------|
| partition | partitioning | 0.597 | -3.94 | 0.114 |
| | dependability | 0.454 | -∞ | 0.037 |
| | android | 0.000 | -5.10 | 0.028 |
| | transportation | 0.213 | -7.17 | 0.033 |
| optimization | optimisation | 5.217 | -4.16 | 0.065 |
| | calibration | 0.542 | -4.47 | 0.079 |
| | internet | 0.000 | -4.16 | 0.010 |
| | innovation | 0.000 | -6.24 | 0.007 |
| classification | categorization | 4.134 | -4.49 | 0.405 |
| | transportation | 0.135 | -3.96 | 0.033 |
| | biometrics | 0.675 | -6.23 | 0.046 |
| | calibration | 1.241 | -5.11 | 0.058 |
| cryptography | steganography | 1.712 | -2.54 | 0.202 |
| | context | 0.408 | -4.37 | 0.035 |
| | innovation | 0.000 | -7.23 | 0.010 |
| | android | 0.662 | -7.20 | 0.096 |
| reliability | dependability | 1.173 | -∞ | 0.157 |
| | integration | 0.483 | -3.11 | 0.023 |
| | forecasting | 0.192 | -5.71 | 0.051 |
| | context | 0.317 | -5.44 | 0.048 |
| cluster | clustering | 4.952 | -4.09 | 0.080 |
| | dependability | 1.724 | -∞ | 0.070 |
| | authorization | 0.000 | -5.14 | 0.056 |
| | correlation | 0.000 | -4.94 | 0.049 |
| contamination | pollution | 0.871 | -3.50 | 0.056 |
| | visualization | 0.068 | -6.02 | 0.021 |
| | entropy | 0.000 | -∞ | 0.020 |
| | sorting | 0.000 | -∞ | 0.007 |
| encoding | encryption | 1.452 | -4.34 | 0.058 |
| | normalization | 0.367 | -4.79 | 0.050 |
| | segmentation | 0.288 | -4.65 | 0.025 |
| | lifetime | 0.412 | -∞ | 0.026 |
| experiment | experimentation | 3.928 | -4.92 | 0.186 |
| | sorting | 0.000 | -5.19 | 0.012 |
| | authentication | 0.000 | -∞ | 0.009 |
| | aggregation | 0.000 | -5.02 | 0.037 |

5 CONCLUSIONS

This paper has presented a new method to measure the similarity between two queries. The experiment for evaluation shows that the effectiveness of this method is quite persuasive by comparing the semantic similarity of synonym and random pairs and finding the synonym among four candidate words. This method could be used to automatically find synonym from a lot of candidate words. It could cope with the changeable web since it uses the web data.

In the future research, the more experiments based on the larger sized dataset should be conducted. Moreover, we will devise the methodology to automatically generate candidate words to find the correct synonym.

ACKNOWLEDGEMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology(2010-0024532)

REFERENCES

- Alqadah, F. & Bhatnagar, R. 2011. Similarity Measures In Formal Concept Analysis. *Annals Of Mathematics And Artificial Intelligence*, 61, 245-256.
- Baroni, M. & Bisi, S. Using Cooccurrence Statistics And The Web To Discover Synonyms In A Technical Language. Lrec, 2004.
- Blondel, V. D. & Senellart, P. P. 2011. Automatic Extraction Of Synonyms In A Dictionary. *Vertex*, 1, X1.
- Cole, R. & Eklund, P. W. 1999. Scalability In Formal Concept Analysis. *Computational Intelligence*, 15, 11-27.
- Curran, J. R. & Moens, M. Improvements In Automatic Thesaurus Extraction. Proceedings Of The Acl-02 Workshop On Unsupervised Lexical Acquisition-Volume 9, 2002. Association For Computational Linguistics, 59-66.
- Ganter, B., Stumme, G. & Wille, R. 2005. *Formal Concept Analysis: Foundations And Applications*, Springer.
- Ganter, B., Wille, R. & Franzke, C. 1997. *Formal Concept Analysis: Mathematical Foundations*, Springer-Verlag New York, Inc.
- Ho, N.-D. & Faron, C. Lexical Similarity Based On Quantity Of Information Exchanged-Synonym Extraction. Rivf, 2004. Citeseer, 193-198.
- Landauer, T. K. & Dumais, S. T. 1997. A Solution To Plato's Problem: The Latent Semantic Analysis Theory Of Acquisition, Induction, And Representation Of Knowledge. *Psychological Review*, 104, 211.
- Lin, D. Automatic Retrieval And Clustering Of Similar Words. Proceedings Of The 17th International Conference On Computational Linguistics-Volume 2, 1998. Association For Computational Linguistics, 768-774.
- Lin, D., Zhao, S., Qin, L. & Zhou, M. Identifying Synonyms Among Distributionally Similar Words. Ijcai, 2003. 1492-1493.
- Lu, Z., Liu, Y., Zhao, S. & Chen, X. Study On Feature Selection And Weighting Based On Synonym Merge In Text Categorization. Future Networks, 2010. Icfn'10. Second International Conference On, 2010. Ieee, 105-109.
- S Nchez, D. & Moreno, A. Automatic Discovery Of Synonyms And Lexicalizations From The Web. Ccia, 2005. 205-212.
- Senellart, P. & Blondel, V. D. 2008. Automatic Discovery

- Of Similarwords. *Survey Of Text Mining II*. Springer.
- Tam, G. K. Focas–Formal Concept Analysis And Text Similarity. Proceedings Of The 2nd International Conference On Formal Concept Analysis, 2004.
- Turney, P. 2001. Mining The Web For Synonyms: Pmi-Ir Versus Lsa On Toefl.
- Van Der Plas, L. & Tiedemann, J. Finding Synonyms Using Automatic Word Alignment And Measures Of Distributional Similarity. Proceedings Of The Coling/Acl On Main Conference Poster Sessions, 2006. Association For Computational Linguistics, 866-873.
- Vechtomova, O. & Wang, Y. 2006. A Study Of The Effect Of Term Proximity On Query Expansion. *Journal Of Information Science*, 32, 324-333.
- Veronis, J. & Ide, N. M. Word Sense Disambiguation With Very Large Neural Networks Extracted From Machine Readable Dictionaries. Proceedings Of The 13th Conference On Computational Linguistics-Volume 2, 1990. Association For Computational Linguistics, 389-394.
- Vickrey, D., Kipersztok, O. & Koller, D. An Active Learning Approach To Finding Related Terms. Proceedings Of The Acl 2010 Conference Short Papers, 2010. Association For Computational Linguistics, 371-376.
- Wille, R. 2009. *Restructuring Lattice Theory: An Approach Based On Hierarchies Of Concepts*, Springer.
- Wormuth, B. & Becker, P. Introduction To Formal Concept Analysis. 2nd International Conference Of Formal Concept Analysis February, 2004.
- Wu, H. & Zhou, M. Optimizing Synonym Extraction Using Monolingual And Bilingual Resources. Proceedings Of The Second International Workshop On Paraphrasing-Volume 16, 2003. Association For Computational Linguistics, 72-79.

A Recommendation System for Specifying and Achieving S.M.A.R.T. Goals

Romain Bardiau, Magali Seguran, Aline Senart and Ana Maria Tuta Osman
Business Intelligence Advanced Development, Sophia Antipolis, SAP Labs, Paris, France
{firstname, lastname}@sap.com

Keywords: Recommendation System, S.M.A.R.T. Goals, Forecast.

Abstract: Businesses and public organizations are typically goal-oriented trying to maximize their performance. Goals are today set arbitrarily and are given without hints on how to achieve them. There are many applications that allow setting up goals and sub-goals but the process is still manual. In this paper, we present a recommendation system that helps the user specify S.M.A.R.T. goals and monitor the progress towards these goals. Given a main goal on a metric, the system recommends specific sub-goals or indicators based on the forecast of historical data. These recommended indicators are the most probable to have a higher contribution in helping the user to reach his main goal. The user can additionally monitor its progress with a visualization over time. We show how this system can be used in a business scenario for sales.

1 INTRODUCTION

Businesses and public organizations are looking for new ways to maximize their precious resources while minimizing costs, especially in difficult economic contexts. Companies' focus is typically on building sustainable competitive advantages around their strategic resources to increase their profit and revenue, regardless the size of their business. Public organizations top concerns are, on the other hand, to improve their service, maximize the use of their resources, and cut expenses.

One effective tool for making progress towards these desired results is to set up goals. Unfortunately, goals are today set arbitrarily and given without hints on how to achieve them. For example, a sale manager may give a goal to his team members like "increase customer satisfaction by 5 points" without any indication on where the effort should be taken, i.e., whether it should be on some specific products or on specific locations. It is up to the employee to find out how he will be able to achieve the goal by digging into and analyzing operational information.

Many applications today can be used for setting, tracking and ultimately achieving goals (Harris, 2013). The user can list his goals and associated tasks, set how often he wants to be reminded of them and can share them with his friends on social networks. However, the process to set up these goals is manual and there is no automated help provided to track their

completeness. Some systems go a step further by dynamically suggesting goal optimizations but they are domain specific and mainly based on user monitoring (Digital, 2013).

In this paper, we present a recommendation system that helps the user specify a S.M.A.R.T. (Specific, Measurable, Assignable, Realistic, Time-related) goal (Meyer, 2003) and monitor the progress towards this goal. The system automatically recommends sub-goals or indicators to the user with minimal interaction. These recommended indicators are the most probable to have a higher contribution in helping the user to reach his main goal. To do so, the system forecasts how the company or organization is likely to perform for several activities based on past operational data and identifies where the effort should be taken to improve the forecasted situation.

The user can additionally monitor the progress of the indicators over time with a visualization widget. This monitoring helps the user discover at a glance the performance achieved. When progress suggests that the goal will unlikely be reached, adjustments of the goal can be made prior to the point in time when the goal should be accomplished.

To illustrate our recommendation system, we will develop a business scenario throughout this paper. We will consider Bob who is a sales representative in Spain. Given the low growth in sales last year for tablets, Bob decides to target a 10% increase in the first trimester of the coming year. We will show how

our solution will recommend indicators to help him reach his goal and will provide him with a widget that he can include on his favorite web page to monitor the progress.

The paper is organized as follows. Section 2 presents the related work. Section 3 presents the recommendation system, while Section 4 describes the architecture and implementation. Finally, Section 5 presents conclusions and future work.

2 RELATED WORK

Recommendation systems are usually using different filtering techniques to find the most interesting items for a user in a large space of possible options (Asanov, 2011). Content filtering techniques take into account the content of items to filter the ones that better match the user's preferences or profile (Kalles et al., 2003). Collaborative filtering is another technique that identifies what items might be of interest for a particular user by looking at the interest of similar users (Sarwar et al., 2001; Herlocker et al., 2004). There are some hybrid works that employs user-based and item-based prediction to guess the rating that a user is going to provide for an item. For example (Papagelis and Plexousakis, 2004) leverages the logged history of ratings and content associated with users and items.

For goal setting, there are many applications that let the user manually set up goals, define sub-goals and monitor the progress from one sub-goal to another sub-goal (Gregory, 2010; GoalScape, 2013; Milestone Planner, 2013) but very few that provide recommendations to the user. Bridgeline Digital (Digital, 2013) proposes a Smart Recommendation Engine that dynamically suggests ways to optimize content for e-commerce. The recommendation engine helps to reach critical campaign goals through web analytics and user monitoring. They have a list of pre-established issues for each goal on which they are running functionality tests (e.g., download link/page not good, landing page does not drive to download).

In comparison, our solution does not need user profiling, logged history (which might lead to privacy issues), or predefined recommendations that usually cannot cover all the possible current and future goals for any domain. Our solution is based on past operational data to generate forecasts from which S.M.A.R.T. recommendations are made. Past operational data might originate from multiple sources and usually designs low-level data such as transactions and prices. To our knowledge, there is no prior work in the literature that uses prediction techniques in order to make recommendations based on forecasted

operational data.

3 RECOMMENDATION SYSTEM

In the rest of the paper, we will assume to be in a business context where operational data is stored in data warehouses. In a data warehouse, data is organized hierarchically according to measures and dimensions. A measure is a numerical fact on operational data. Examples of measures for specific product sales data include quantity sold, revenue, and profit margin. Each measure can be categorized into one or more dimensions. Dimensions define categories of stored data. Examples of dimensions include time, location, and product. Each dimension can have one or more sub-dimensions. For example, the time dimension can include sub-dimensions of year, quarter, month, week, and so on.

3.1 Business Workflow

This section presents the business workflow detailing the different steps that are followed to generate goal recommendations.

3.1.1 Main Goal Setting

The user first defines a goal for a target measure by providing input to a GUI, as depicted in Figure 1. A quantitative value as point or percentage target needs to be entered for the goal. The user then selects the orientation of the defined goal (e.g., increase, decrease, maintain) from a drop down menu and specifies the time frame by selecting the dates in a calendar. The user finally selects a measure and possibly dimensions for the goal. The measures can be retrieved and ranked according to contextual data retrieved from a user profile (e.g., job, title, location) or with collaborative filtering techniques on historical data associated with the user (Liu et al., 2012). The user can alternatively select one measure from the set of all available measures in the data warehouse. The dimensions will act as filters and will restrict the scope of the selected measure.

Within the example business scenario discussed above, Bob can provide user input to define a goal for increasing sales growth by 10% from January 2014 to March 2014. More particularly, Bob can select the measure "sales growth" from the recommended measures, input "10" for the percentage target, select "increase" and set start and end dates. In Figure 1, the measures "margins", "revenue" and "total sales"

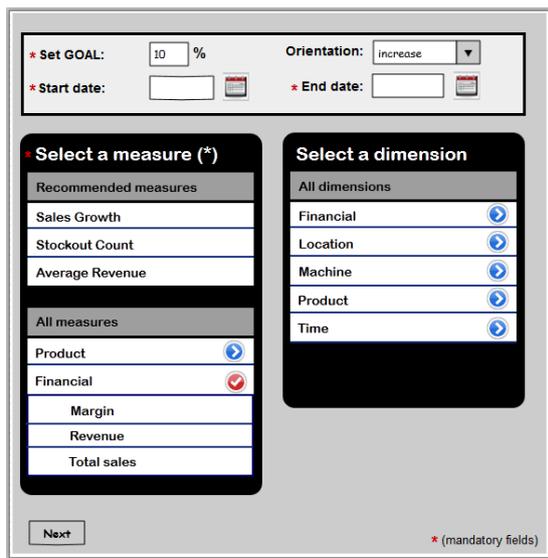


Figure 1: Goal Setting.

were also available for selection in the "financial" category of the data warehouse. After providing his input, Bob can select the "Next" button to progress to recommended indicators.

3.1.2 Recommended Indicators

Once the user has defined his main goal, the system finds and recommends dimensions and sub-dimensions for the selected measure and associated filters. Figure 2 depicts an example GUI for displaying these recommendations and for enabling user selection of one or more recommendations. These recommendations are ranked based on the risk of the goal to be reached as it will be later explained in Section 3.2.

The user can select a dimension from the recommended dimensions selection menu and/or from the all dimensions menu. The dimensions listed in the latter menu include all available dimensions in the data warehouse. In the scenario, the recommended dimensions include dimension and sub-dimension "city;Barcelona" having a rank of 4.4 and other dimensions and sub-dimensions with a lower rank.

The combination of the selected measures, filters and dimensions define the indicator, which can be monitored over the selected time frame to assist the user in achieving the defined goal.

3.1.3 Goal Monitoring

Once an indicator has been fully specified by the user, the system generates a software widget that monitors

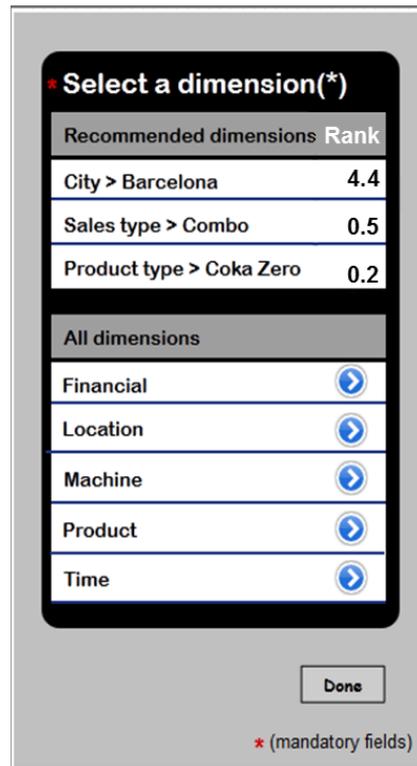


Figure 2: Selection of a Recommended Indicator.

the progress of the selected indicator towards reaching the goal (see Figure 3). A widget is a visual application or component comprising portable code intended to be used on different platforms. In addition, alerts are automatically set to inform the user when the trend has an undesired direction or reach a critical level, depending on user's preferences. The alert can support the user to perform an action at a critical time in the timeframe to decrease the risk not to reach the goal. Note that the recommendation of actions is out of the scope of this paper.

In our scenario, if the user has selected "City;Barcelona" as indicator, then the system creates a widget to monitor the trend of the sales growth in Barcelona over time with alerts when the growth decreases. Bob adds the widget on his portal and enterprise blog to track the sales growth and share it with his colleagues.

3.2 Recommendation Model

Our recommendation model is based on a small set of concepts that are defined in this section.

First, we define *risk* as the probability of the recommended indicators to be above or below a certain threshold in the future. A forecast of operational data is performed to determine this risk. The *threshold* can be based on one or more known trends associated with

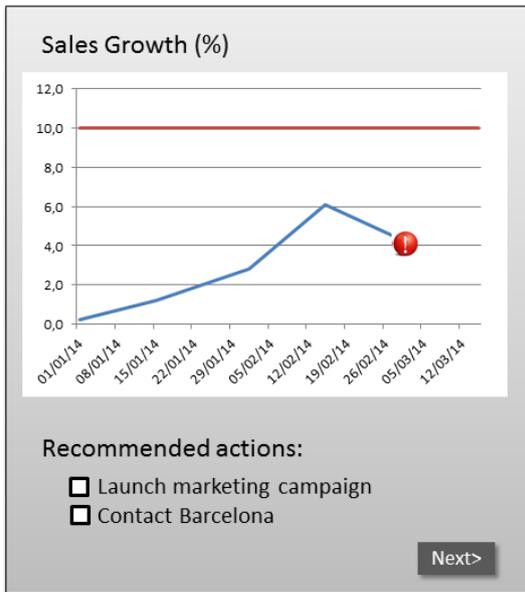


Figure 3: Widget alert.

a measure. For example, the threshold can be the minimum value or a maximum value that is established for an indicator. In our case, the threshold is the value associated with the goal defined by the user (e.g., 10% in our scenario). The risk can be determined using a cumulative distribution function (CDF) based on the predicted trend and the threshold as follows:

$$risk = CDF(\text{futureTrend}, \text{threshold}) \quad (1)$$

We define the *contribution* of an indicator as the relative importance of the indicator in reaching the selected goal. The contribution can be specified manually by setting weights for indicators, or it can be automatically calculated. In our system, the contribution is calculated based on a predicted trend and one or more known trends:

$$contribution = \frac{(\text{futureTrend} - \text{pastTrend})}{\sum(\text{futureTrend} - \text{pastTrend})} \quad (2)$$

Finally, the rank of an indicator can be determined based on a level of risk associated with the indicator and the contribution in reaching the defined goal:

$$rank = risk * contribution \quad (3)$$

4 IMPLEMENTATION

In this section, we will present the general architecture and then how it has been implemented on SAP HANA Appliance product (Farber et al., 2011).

4.1 Architecture

Figure 4 depicts the general architecture of the solution that is composed of three layers: a GUI layer, an engine layer, and a database layer. The GUI layer on the client-side provides interfaces from Section 3.1 that can be displayed on any computing devices (desktop, laptop, smartphones, etc.). The engine layer and the database layer are hosted on one or more servers. Communication between the client and the servers is over HTTP.

4.1.1 GUI Layer

The user interacts with a GUI of the GUI layer to generate input data that is provided to the engine layer. User input includes at least an operational goal, a timeframe, an orientation, a measure and a dimension.

4.1.2 Engine Layer

The engine layer contains the following components: an engine for storing user input, a forecast engine, a risk calculator, a contribution calculator, and a rank calculator. Data is automatically retrieved (e.g. at particular intervals) or selectively retrieved by one or more engines of the engine layer from one or more sources, processed and stored in the database layer.

Store User Input. This component receives data that has been input through the GUI and provides the data for storage in the user input data store. The user input data will be later consumed by several engines of the engine layer.

Forecast Engine This component receives user input data and past operational data (e.g., stored in the past operational data store). Based on a prediction algorithm, it processes the user input data and the past operational data to generate a forecast that will be stored in the forecast data store.

Risk Calculator. This component determines a risk associated with the forecast based on the goal and orientation provided from the user input. The risk is stored in the dictionary data store.

Contribution Calculator. This component receives user input data and forecast data and processes the contribution associated with a respective indicator. The contribution is stored in the dictionary data store.

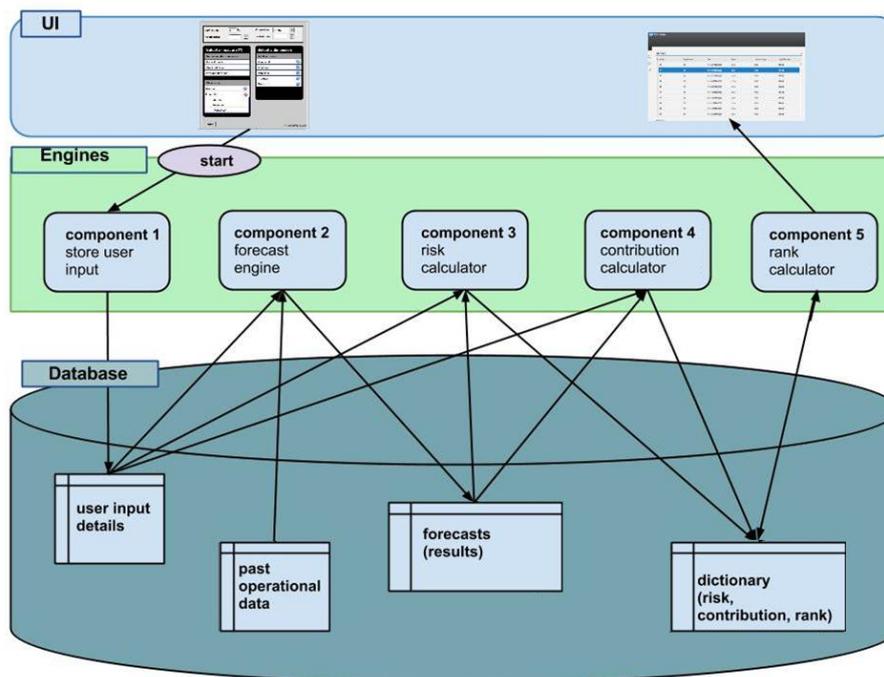


Figure 4: Implementation.

Rank Calculator. This component determines the rank of an indicator based on the level of risk associated with the indicator and the contribution in reaching the defined goal. The rank calculator stores the determined rank in the dictionary data store. One or more indicators and respective ranks can be displayed to the user in the GUI.

4.1.3 Database Layer

The database layer includes user input data store, past operational data store, forecast data store, and dictionary data store. These stores provide all the necessary tables.

4.2 Implementation in SAP HANA

We have developed the GUI layer with SAPUI5, a UI Development Toolkit for HTML5 (Network, 2013). This framework offers a series of libraries that front-end developers can use to build compelling HTML5-based applications. The recommendation service has been implemented as a native application running on the extended application services server (XS server) of SAP HANA Appliance for better performance. The native application calls two stored procedures running at the database level. The first one is a SQL procedure for calculating the forecasts based on the exponential smoothing algorithm from the Predictive Analytics Library of HANA. The second stored pro-

cedure calculates the risk and the contribution with R code executed in the SAP HANA database query execution plan. The input for these two stored procedures is found in a SAP analytical view and the output is stored for future retrieval in relational tables in the SAP HANA in-memory database system. Whenever the user is calling the UI for recommendations, SAPUI5 uses an OData service (Kirchhoff and Geihs, 2013) for querying the recommendation service.

5 CONCLUSION

The system presented in this paper recommends one or more indicators to be monitored for reaching a goal. More specifically, it enables users to define accurate and realistic goals, and supports user monitoring of the progress toward goals based on the one or more indicators that have been recommended. Indicators point to areas of improvement and can act as triggers for action for the user. The novelty of our approach is based on forecasting the past operational data for finding unfavorable trends in the future and make recommendations based on them.

In the future, we plan to introduce machine learning algorithms to provide better recommendations to the user based on the usage of our recommendation system. We would also like also to leverage the ERP systems to automatically propose templates for differ-

ent business lines.

REFERENCES

- Asanov, D. (2011). *Algorithms and Methods in Recommender Systems*. Berlin Institute of Technology.
- Digital, B. (Accessed in 2013). iAPPS Analyzer. <http://www.bridgelinedigital.com/website-management/web-analytics-solutions>.
- Farber, F., Cha, S. K., Primsch, J., Bornhovd, C., Sigg, S., and Lehner, W. (2011). SAP HANA Database: Data Management for Modern Business Applications. In *SIGMOD Rec.*, volume 40, pages 45–51.
- GoalScape (Accessed in 2013). GoalScape: Visual Goal Management Software. <http://www.goalscape.com>.
- Gregory, A. (2010). Goal Setting tracking tools. <http://www.sitepoint.com/goal-setting-tracking-tools>.
- Harris, J. (Accessed in 2013). Crucial tools for business goal setting. <http://www.paydayhr.com/1/post/2013/02/3-crucial-tools-for-business-goal-setting.html>.
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22:5–53.
- Kalles, D., Papagelis, A., and Zaroliagis, C. (2003). Algorithmic aspects of web intelligent systems. In Zhong, N., Liu, J., and Yao, Y., editors, *Web Intelligence*, pages 323–344. Springer Berlin Heidelberg.
- Kirchhoff, M. and Geihs, K. (2013). Semantic description of odata services. In *Proceedings of the Fifth Workshop on Semantic Web Information Management*, number 2 in SWIM 13, pages 2:1–2:8, New York, NY, USA. ACM.
- Liu, Q., Chen, E., Xiong, H., Ding, C., and Chen, J. (2012). Enhancing collaborative filtering by user interest expansion via personalized ranking. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(1):218–233.
- Meyer, P. J. (2003). *Attitude Is Everything: If You Want to Succeed Above and Beyond*. Meyer Resource Group, Incorporated.
- Milestone Planner (Accessed in 2013). Milestone Planner: Powerfully Simple Planning. <http://milestoneplanner.com>.
- Network, S. C. (Accessed in 2013). UI Development Toolkit for HTML5 Developer Center. <http://scn.sap.com/community/developer-center/front-end>.
- Papagelis, M. and Plexousakis, D. (2004). Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents. In *Cooperative Information Agents VIII*, volume 3191 of *Lecture Notes in Computer Science*, pages 152–166. Springer Berlin Heidelberg.
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, pages 285–295, New York, NY, USA.

An Self-configuration Architecture for Web-API of Internet of Things

Eric Bernardes Chagas Barros and Admilson de Ribamar L. Ribeiro
Computing Department, Universidade Federal de Sergipe, Aracaju, Brazil
bernardes.eric@gmail.com, admilson@ufs.br

Keywords: IoT, Web-APIs, Self-configurable, Internet of Things, Web of Things, WoT, Self-configuration, REST, Mote.

Abstract: The internet of things (IoT) is the paradigm that will dominate the computing world in the coming years. In this way, studies should be conducted in such way to ensure its enhancement and in the quest for that improvement is necessary to use the already existing technologies that apply to IoT. This paper's purpose is to unite different technologies like REST, cloud computing and embedded operating system in order to obtain mechanisms capable of self-configuration. Thus, it was possible to conclude that the architecture proposed would increase useful techniques for the implementation of systems that want to run the self-configuration as well as assist in setting up networks of computers that work with wireless sensors and IoT.

1 INTRODUCTION

Internet of Things (IoT) is the paradigm that until 2025 will dominate the world of computing (Atzori, 2010). The ubiquity of the Internet in less than 20 years almost connected all people in the world and has generated new demands for space. Now people not only need to exchange information and services, but also objects. Although for this to occur it is still necessary that both, the technology and society are prepared.

The Things which refers to this paradigm are related to devices or motes that are arranged in an environment and have their own characteristics, how to measure a temperature, check if the light is on, if a window is open, the amount of milk in a refrigerator, among many other possibilities. By capturing the information for which they were programmed, these devices send data through existing Internet's services to let user become aware that these data are stored and can provide some useful information to him. The use of existing resources on the web by these motes can be done through APIs and can be characterized by the use of cloud computing, when this happens it is customary to call these APIs of Web-APIs (Zeng, 2011).

The configuration and installation of devices that will integrate a large and complex systems within the IoT is a challenge that is time consuming and error prone, even for the great specialists (Kephart,

2003). Moreover, the large growth of network nodes made with different technologies and different platforms can result in a hard and repetitive work.

Therefore, to facilitate the use of such devices, and applications' development, the use of techniques that enables the system to adapt itself to the environment and self-configure is a great need. However, the Web-APIs that exists in nowadays has not yet incorporated these concepts yet.

This paper aims to introduce a mechanism of self-configuration for the Internet of Things, where the main idea is to make easier the configuration of devices and Web-APIs that will control the environment.

This paper aims to introduce a mechanism of self-configuration for the Internet of Things, where the main idea is to make easier the configuration of devices and Web-APIs that will control the environment.

The remainder of this paper is organized as follows. In section 2, will be shown on the existing requirements nowadays and that contribute to the development of Web-APIs. Section 3 talks about the Web-APIs that exists in the market and the summary of its main characteristics. Finally, in Section 4, the proposed architecture will be explained and a possible mechanism architecture that can be used in the development of self-configurable Web APIs. Conclusion and future research hints are given in Section 5.

2 REQUIREMENTS OF WEB-APIS IN INTERNET OF THINGS

Currently, the existing Web-APIs have a set of basic characteristics that are used to carry out the communication of devices with the Internet or for needs of these nodes due to its limitations. These fundamental features are described in this article in order to enumerate some of the concepts that can be used to serve as basis for self configuration mechanism, such as the form of communication with Rest (Zeng, 2011), storage and standardization communication through the use of markup data languages (XML, YAML, JSON) (Xively 2013).

2.1 Open-source

Although this characteristic is not a specific functionality that help directly the devices, it was regarded as important for that in the future people will work on top of existing Web-APIs and make your code to be improved and become Customer self-configurable.

This term refers to the so-called free software, where to be held a consolidated distribution, is also distributed its source code for that can be freely used, modified and shared by its users.

2.2 Rest

The REST-based architecture is considered "the true architecture of the Web" (Zeng, 2011), it is based on the concept that everything is modeled as resource using the HTTP URI. Thus, customers can identify the resources they need through the URI, manipulating them through traditional HTTP commands like: PUT, GET, POST and DELETE. The PUT and DELETE.

Moreover, it has self-descriptive messages, i.e, the resources are free to make their own representations of data format. Obviously, end-systems must agree with this representation so that communication can take place properly. In this way, it is possible to use HTML, XML, text, PDF and images as the format of data to be sent.

Another important feature is that REST works with stateless requests, treating each request independently, and this may not require a server to store session information or the status as is each of the multiple acquisitions. However, statefull interactions can be supported in REST through the use of hyperlinks, so the states of the resources can

be transferred by means of URIs for cookies or hidden fields (Zeng, 2011).

2.3 Standardization

As the APIs and the devices are usually developed in different languages, it must be pre-established a format of data communication between the receiver and transmitter and how they will exchange messages to inform how the data is separated and what the content within it represent. Consistently, to earn this type of representation the IoT sought markup languages known data, such as XML, JSON, YAML or CSV.

These languages are very portable because it does not depend on hardware or software platforms to work and any databases can communicate with each other through them. By having the ability to self define data, as well as having the characteristics described above, these languages are used for interoperable networks, allowing objects of different characteristics understand each other.

2.4 Centralized Architecture

Due to the limitations of the devices many of the activities more robust need to be sent to a server that has capacity to perform a greater load of processing and storage. Therefore currently the Web-APIs, tend to be centered on a server that is able perform this type of activity. Thus, a network IoT using these Web-APIs tend to use the REST to communicate with a server that is receiving data and managing the devices in the network.

2.5 Security

When the term security is mentioned, the first word illustrated is identification. In IoT, recognition of each device with the use of traditional IPs. Despite this, only a network identification is not sufficient to ensure the safety, it is necessary a profile control to inform if this equipment has access to the service that it is requesting. As in IoT these services are provided by APIs, the controls of inflows are usually made by API-Keys.

Within the API-Key are encapsulated three types of permissions that operate in a hierarchical manner: object key (the general key of the API), object permissions and the permissions of features of objects, the latter being optional. The general permissions objects keys are created for your applications to have access to APIs. Each application may ask how many objects keys you

need.

An object key can have multiple objects permissions (it is mandatory at least one) and each acquiescence of object contains a set of different permissions. For example, a key can be created to allow a read-only access to the entire public resource available, in the same way, can allow a write access to a resource responsible for supply of data by means of a specific IP (Xively, 2013). There are still the feature permissions that serve to restrict access to a given resource. These permissions, as were elucidated above, are optional.

In addition to the API-Keys which grants the security level of access control, there is the HTTPS that provides security at the level of sending and receiving data throughout an encrypted and secured channel. As the APIs' principal way of communication focus on HTTP (REST and SOAP) the use of HTTPS helps to prevent the attacks of type man-in-the-middle, seen that the HTTPS is the implementation of HTTP on top of the SSL/TLS protocol to provide authentication of hosts purposes and encrypted communication between them.

2.6 Self-configuration

With the advance of network technologies, devices shipped and software tools, the growth of heterogeneous nodes, of great complexity and extremely dynamic that pass to operate within the Internet becomes something that cannot be easily administered.

The autonomic computing is inspired in the human being's nervous system. Its main objective is to develop applications that can self-according to guidelines imposed by human beings at a high level. Thus with the policies established at a high level it is possible to make the systems self-reliant to self-configure, self-healing, self-optimization and the self-protection (Kephart, 2003).

The self-configuration is responsible for automated configuration of system components, with it the system will automatically adjust and it always will adjust based on policies of self-configuration. The self-optimizing components and systems continually seeking opportunities to improve their own performance and efficiency. Self-healing the system automatically detects, diagnoses and repairs problems of software and hardware located. The Self-Protection system automatically if defends against malicious attacks or cascading failures. It uses earlier warnings to anticipate and prevent failures in the entire system (Kephart, 2003).

2.7 Code-source Device

As each device needs to communicate with the Web-API through the REST, many of these offer codes-sources for which the user copy and paste in Integrated Development Environment (IDE) responsible for programming the device. In this way it is possible to at least have an example that how to program a device and if it is possible already find a code that is applicable to the device that will enter the network.

However it is important to realize that although there may be a useful source code available for the used equipment perform the copy and paste codes for multiple appliances can be an arduous task and subject to errors even for the great specialists, once that may exist dozens of these to be configured in a single environment.

2.8 Storage

The storage of data in IoT is an interesting area of study, once the majority of devices that exist in a network of this type do not have large storage capacity. In contrast, the data used in the communication are stored in a central device that has the characteristics necessary for the recording of relevant data to the system.

As the WEB-APIs are on a server that contains high processing power and storage capacity, they are usually responsible for the storage of data that is captured and transmitted by devices. For this reason, it is used the concept of Feeds (system risers). These feeders are a specific part of the API that works with the reading and writing of data from the system.

Each Feed is a set of channels (datastream) that enables the exchange of data between the APIs and authorized devices. These channels are designed by programmers to separate the data by specific characteristics. In view of this, it is possible to create public and private channels. The first are those that can be viewed and changed by all according to the BCC License, already the second, it is those whose access is permitted only to developers and those whose admission is granted. Within channels there is the concept of DataPoint which is the representation of the data in a given time (timestamp) (Xively, 2013).

3 RELATED WORK

In this section will be elucidated the main existing Web-APIs and what features they have. These

features are related to the requirements that were previously seen.

3.1 ThingSpeak

ThingSpeak is an API for "Internet of Things" open-source that stores and retrieves data from devices using the Hypertext Transfer Protocol (HTTP) over the Internet or simply of a LAN (Local Area Network). With this API (Application Programming Interface) it is possible to create applications in sensors for data records in a given environment, tracking, location and social networks of "things" (ThingSpeak, 2013). The data manipulation occurs by means of its channels, which have eight fields to be fed with data numeric and alphanumeric pagers, in addition to fields such as latitude and longitude, elevation and status.

3.2 NimBits

It is a collection of software components designed to record data of time series, such as for example, the changes in temperature read by a given sensor (NimBits, 2013). This API has the drive of events (triggers) during the recording of data. In This way, it is possible to perform calculations or trigger alerts along with your receipt.

Another advantage is that it was designed to be the first historian of world data, which means that you can download it and install it on any server, local or in the cloud, so that it is used the Linux Ubuntu and Google App Engine. This approach allows all instances of API to relate being possible to find other feeders (feeds) of data and make possible a connection with them (NimBits, 2013).

3.3 Open.sen

Currently in beta stage, this tool allows a rich visualization of results, so that by SenseBoard, you can see the incoming data in real time. The SenseBoard is powered by applications that are developed and installed within the API itself. These applications are independent but can be easily integrated with feeder (feeds) devices (Open.sen., 2013). These feeders communicate with the API through channels that are connected to devices. Even so, it is possible to capture information from other applications, not needing a direct contact with the device.

3.4 Cosm

This tool (formerly called Pachube) was developed to be a platform as a service (PaaS) for the Internet of things. With it, you can manage multiple devices through the RESTful resources, thus it is possible to deal with all the components of the API (Feeds, triggers, datastreams and datapoint) using commands via HTTP URLs, as already seen, PUT, GET, DELETE, POST. PUT is used to change the data, GET to reading, DELETE to erase and POST to create resources to communication or control (Xively, 2013).

3.5 SensorCloud

The SensorCloud is a tool storage for sensors "things". SensorCloud provides a Rest API to allow the upload of data to the server. The API implementation is based on patterns of HTTP commands. Soon, it is easily adapted to any platform (SensorCloud, 2013).

The communication between the tool and the sensors is totally on top of HTTPS, which means the entire communication between the channels and the devices are encrypted.

The format for sending and receiving data is the XDR (External Data Representation), it is not yet possible to use the templates known JSON and XML. The purpose of not using the formats standards of delivery is due to the fact that the XDR is not text but binary mode and with this it is possible that sensors for low processing power are able to send larger amounts of data than the standards based on text (SensorCloud, 2013).

Their components are divided hierarchically, where the device is at a higher level and contains the sensors that are divided into channels and these have the data.

3.6 Evrythng

It is a platform for powering applications or services directed by dynamic information about physical objects. Your goal is that all things must be connected, thus sets a world where all 'Thng' have a digital presence of assets on the Internet, even in social networks if desired, allowing the rapid development of Web applications using real-time information flowing from, any object in the world (Evrythng 2013).

3.7 iDigi

It is a platform in the cloud for managing network devices. It offers management gateways and endpoints on the network. It presents security policies of leaders in the industry, and great scalability for the exponential growth of devices on the network (Etherios, 2013).

3.8 GroverStream

GroveStreams is one of the most powerful platforms in clouds capable of providing real-time decision making for millions of users and devices. Among several of its qualities is the code generation per device. In this API it is possible that when you choose your device and the function that it will play a code that can be used to synchronize the device with the API is generated, thus there is only a need to copy this code paste in compiler used by the device and send to motto for which the code was generated (GroverStream, 2013).

3.9 Comparison of Web-APIs

Until the moment when it was explained the main features of Web-APIs, however as it can be seen in Table 1, there are some gaps that are still not handled by any of them. One of these characteristics is the autoconfiguration. Although, many Web-APIs provide examples of codes for configuring devices, none of them provides a side focused for the weak link in the IoT, the motes.

4 PROPOSED ARCHITECTURE

The mechanism proposes an architecture that

addresses two types of problems. The first is the reduction of the complexity of devices' initial configuration, currently this setup is done through the provision of source code on the part of the Web-API and the compilation and deploying into the device by the user that is configuring the network. Then there is he must to go to the Web-API and configure it to receive the data according to the settings that were made available for the device.

The second problem is the reconfiguration of device that have already been configured and deployed on the network. Actually, if it is required reconfigure some devices that are already doing their job, the user needs to go up to the place where the mote is and remove it from the network to perform again the setup process, passing again by the first problem that was quoted.

The proposed mechanism will be divided into two parts: CLIENT and SERVER. Figure 1 provides a representation of the architecture.

The CLIENT's side Web-API will be native in motes and will be developed using techniques for low power consumption and memory. Initially intends to use the C language that can be compiled in almost all devices and techniques of concurrent programming and events: Protothread. The project will be carried out using Contiki, because in addition to being lightweight and perform the treatment of energy control of wireless transmitters through the ContikiMAC, it can also be run in almost all existing devices.

Thus, the CLIENT will provide to the SERVER's side the initial information of itself: what it is, what type of service it offers, what pin he uses to read, etc. For example, to connect a device into a network, it goes in search of Web-API for which was previously configured. When found, it sends the same to the information it holds about the device on which it is hosted. For realization of this

Table 1: Characteristics of Web-APIs.

| | Open-Source | REST | Markup language data | Centralized Architecture | Security | Self Configuration | Provides Code for the device configuration | Cloud Storage |
|--------------|-------------|------|----------------------|--------------------------|----------|--------------------|--|---------------|
| ThingSpeak | x | x | x | x | x | | x | x |
| Nimbits | x | x | x | x | x | | x | x |
| Open.sen | | x | x | x | x | | x | x |
| Cosm | | x | x | x | x | | x | x |
| SensorCloud | | x | | x | x | | x | x |
| Evrythng | | x | x | x | x | | x | x |
| iDigi | | x | x | x | x | | x | x |
| Grovestreams | | x | x | x | x | | x | x |

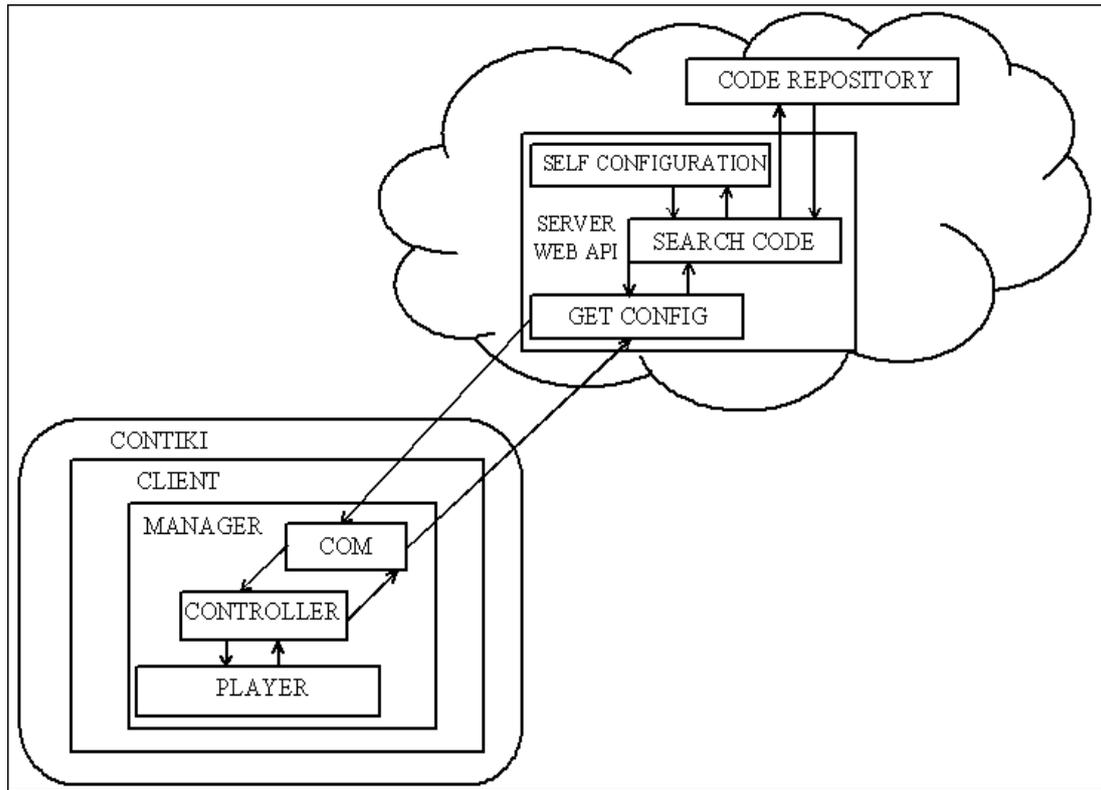


Figure 1: Proposed Architecture.

communication standard message format is also proposed.

The CLIENT is divided into 4 parts:

- Responsible for run of the code (Player);
- Responsible for version of configurations that are rotating (Controller);
- Responsible for sending and receiving of configuration data (COM);
- General monitoring of all parties (Manager).

After receive the source code COM passes for the CONTROLLER that will check the version and validations necessary for the implementation of the code, if it is accepted it goes to the PLAYER who will run the code received. If not, it creates an invalid code message and returns to the COM, which will send to the SERVER.

The Web-API that receive the information will check the authentication of this device (APIKEY) and the type of configuration that it needs to be able to carry out their functions.

As there are many repositories available for devices within the Internet, as the github, the initial idea is to go after the existing codes and transfer them to the requesting device. Thus in order to go search code of a specific device, the WEB-API first must receive a GET with the specific characteristics of the device.

Upon receiving the search engine of the desired settings it returns to the mote, which will start the process of self-configuration.

When the configuration that will be passed to the device is discovered, the Web-API will also need to self-configure, providing new features to the code that is being executed by the motes can send the information. Even so, through these features developers can also configure new applications in the cloud that can promote the environment, such as sharing data between Web-APIs, and integration with social networks.

5 CONCLUSIONS

The architecture presented will initially provide two benefits: First is the development of techniques that may be useful for systems that want to implement the self-configuration.

The second one is the own development a tool using the architecture, once implemented this mechanism, it will assist in setting up a computers' network that works with wireless sensors and IoT. Thus, if the focus is the analysis of new systems of sensors, configuration will no longer be a complicated and a

time consuming step, letting all the attention be directed to the main objective of the research.

For future work there is the possibility of developing the proposed architecture using existing Web-APIs, stated that work like ThingSpeak and Nimbits are great candidates for this development since they have available for IoT good features and they are open-source.

Another challenge is the development of a Web-API that will provide the self-configuration for other existing Web-APIs. With this, besides the implementation of the proposed architecture for IoT, will need to draw up a technical architecture and interoperability for the Web-APIs of the Internet of Things and the clouds.

REFERENCES

- Atzori, L; Iera, A; Morabito, G. The Internet of Things: A survey. *Computer Networks*, 2010. Computer Network.
- Bandyopadhyay, S. A. Survey of Middleware for Internet of Things. *International Journal of Computer Science & Engineering*. Survey (IJCSSES), 2011.
- Zeng D., Guo S, and Cheng Z. The Web of Things: A Survey. *Journal of Communications*, vol. 6, setembro 2011.
- <https://xively.com/dev/docs/api/security/keys/> last accessed October 14, 2013.
- <http://www.json.org/> last accessed October 14, 2013.
- <http://www.w3.org/XML/> last accessed October 14, 2013.
- http://www.computerworld.com/s/article/43487/Application_Programming_Interface last accessed October 14, 2013.
- <https://www.thingspeak.com/> last accessed October 14, 2013.
- <http://www.nimbits.com/> last accessed October 14, 2013.
- <http://open.sen.se/apps/29/> last accessed October 14, 2013.
- http://www.sensorcloud.com/sites/default/files/SensorCloud_Open_Data_API.pdf last accessed October 14, 2013.
- <https://grovestreams.com/> last accessed October 14, 2013.
- <http://www.etherios.com/> last accessed October 14, 2013.
- <http://www.contiki-os.org/> last accessed October 14, 2013.
- Ruane, Laure. Protothread. UserGuide. <https://code.google.com/p/protothread/wiki/UserGuide>. 2013 last accessed October 14, 2013.
- Parachar M, Hariri S. Autonomic Computing: An Overview. *Springer Berlin Heidelberg*. 2005.
- Kephart, Jeffrey O. The vision of Autonomic Computing. *IEEE Computer Society*. 2003.
- <http://www.evrythng.com/> last accessed October 14, 2013.

A Comparison of Three Pre-processing Methods for Improving Main Content Extraction from Hyperlink Rich Web Documents

Moheb Ghorbani¹, Hadi Mohammadzadeh² and Abdolreza Nazemi³

¹*Faculty of Engineering, University of Tehran, Tehran, Iran*

²*Institute of Applied Information Processing, University of Ulm, Ulm, Germany*

³*School of Economics and Business Engineering, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany*
moheb.ghorbani@ut.ac.ir, hadi.mohammadzadeh@uni-ulm.de, abdolreza.nazemi@kit.edu

Keywords: Main Content Extraction, Pre-processing Algorithms, Hyperlink Rich Web Documents.

Abstract: Most HTML web documents on the World Wide Web contain a lot of hyperlinks in the body of main content area and additional areas. As extraction of the main content of such hyperlink rich web documents is rather complicated, three simple and language-independent pre-processing main content extraction methods are addressed in this paper to deal with the hyperlinks for identifying the main content accurately. To evaluate and compare the presented methods, each of these three methods is combined with a prominent main content extraction method, called DANAg. The obtained results show that one of the methods delivers a higher performance in term of effectiveness in comparison with the other two suggested methods.

1 INTRODUCTION

A huge volume of web pages being mainly text is placed on the web every day. A significant proportion of this data is published in news websites like CNN and Spiegel as well as information websites such as Wikipedia and Encyclopedia. Generally speaking, every web page of the news/information websites involves a main content (MC) and there is a great interest to extract it at a high accuracy because the MC can be saved, printed, sent to friends and etc. thereafter.

In spite of the numerous studies which have been done during the recent decade on extraction of the MC from the web pages and especially from the news websites, and although many algorithms with an acceptable accuracy have been implemented, they have rarely paid attention to two critical issues, namely pre-processing and post-processing. Thus, these MC algorithms were not fully successful in some cases. Particularly, the MC extraction algorithms have often failed to extract the MC from the web pages which contain a great number of hyperlinks like for example Wikipedia. This paper will introduce and compare three different methods which can be used for pre-processing of the MC extraction algorithms based on HTML source code elements. Each of the three presented methods is combined with a DANAg (Mohammadzadeh et al., 2013) algorithm as a pre-processor

in order to be able to compare them with each other. The obtained results show that one of the suggested methods is very accurate.

This paper is organized as follows: Section 2 reviews the related work briefly, while the pre-processing approaches are discussed in Section 3. The data sets and experiments are explained in Section 4, and Section 5 makes some conclusions.

2 RELATED WORK

Algorithms and tools which are implemented for main content extraction usually employ an “HTML DOM tree structure” or “HTML source code elements” or in simple words HTML tags. Algorithms can also be divided into three categories based on the HTML tags including “character and token-based” (Finn et al., 2001), “block-based” (Kohlschütter et al., 2010), and “line-based”. Most of these algorithms need to know whether the characters in an HTML file are components of content characters or non-content characters. For this purpose, a parser is usually used to recognize which type of the component they are. Character and token-based algorithms take an HTML file as a sequence of characters (tokens) which certainly contain the main content in a part of this sequence. Having executed the algorithms of this section, a sequence

of characters (tokens) is labeled as the main content and is provided to the user. BTE (Finn et al., 2001) and DSC (Pinto et al., 2002) are two of the state-of-the-art algorithms in this category. Block-based main content extraction algorithms, e.g. boilerplate detection using shallow text features (Kohlschütter et al., 2010), divide an HTML file into a number of blocks, and then look for those blocks which contain the main content. Therefore, the output of these algorithms is comprised of some blocks which probably contain the main content. Line-based algorithms such as CETR (Weninger et al., 2010), Density (Moreno et al., 2009), and DANAg (Mohammadzadeh et al., 2013), consider each HTML file as a continuous sequence of lines. Taking into account the applied logic, they introduce those lines of the file which are expected to contain the main content. Then, the main content is extracted and provided to the user from the selected lines. Most of the main content extraction algorithms benefit from some simple pre-processing methods which remove all JavaScript codes, CSS codes, and comments from an HTML file (Weninger et al., 2010) (Moreno et al., 2009) (Mohammadzadeh et al., 2013) (Gottron, 2008). There are two major reasons for such an observation: (a) they do not directly contribute to the main text content and (b) they do not necessarily affect content of the HTML document at the same position where they are located in the source code. In addition some algorithms (Mohammadzadeh et al., 2013) (Weninger et al., 2010) normalize length of the line and, thus render the approach independent from the actual line format of the source code.

3 PRE-PROCESSING METHODS

In this section, all kinds of the pre-processing methods are explained in detail. Hereafter, these methods are referred to as Filter 1, Filter 2, and Filter 3, for further simplicity. In this contribution, only the presented pre-processing methods are combined with one of the line-based algorithms which is called DANAg (Mohammadzadeh et al., 2013).

3.1 Filter 1

Algorithm 1 shows the simple logic used in Filter 1. It can be seen that one just needs to remove all the existing hyperlinks in an HTML file which is done at line 4 of this algorithm. The only disadvantage of this pre-processing method is that by removing the hyperlinks, the anchor texts are also removed. As a result, this will cause the hyperlinks in the extracted main content to be lost. Thus, their anchor texts, which

must be seen in the main content, will no longer exist in the final main content. Consequently, the application of Filter 1 will reduce either the accuracy or the amount of recall (Gottron, 2007). In the ACCB algorithm (Gottron, 2008), as an adapted version of CCB, all the anchor tags are removed from the HTML files during the pre-processing stage, i.e. Filter 1.

Algorithm 1: Filter 1.

```

1:  $Hyper = \{h_1, h_2, \dots, h_n\}$ 
2:  $i = 1$ 
3: while  $i \leq n$  do
4:    $h_i.remove()$ 
5:    $i = i + 1$ 
6: end while

```

3.2 Filter 2

The idea behind Filter 2 which is shown in Algorithm 2 implies that the all attributes of each anchor tag are removed. With respect to Algorithm 2, which shows the pseudocodes of Filter 2, one can understand that an anchor tag contains only an anchor text.

```
<a>anchor text</a>
```

An advantage of Filter 2 over Filter 1 is that some anchor texts related to the anchor tags, which are located in the main content area, can be extracted by using Filter 2. In other words, the amount of recall (Gottron, 2007) yielded from application of Filter 2 would be greater than the one obtained from Filter 1.

Algorithm 2: Filter 2.

```

1:  $Hyper = \{h_1, h_2, \dots, h_n\}$ 
2:  $i = 1$ 
3: while  $i \leq n$  do
4:   for each attribute of  $h_i$  do
5:      $h_i.remove(attribute)$ 
6:   end for
7:    $i = i + 1$ 
8: end while

```

3.3 Filter 3

In the third pre-processing method, called Filter 3, all the HTML hyperlinks are normalized. In other words, the purpose of this method is to normalize the ratio of content and code characters representing the hyperlinks. Filter 3 is addressed in the AddDANAg (Mohammadzadeh et al., 2012) algorithm.

For further simplification and better comprehension, the underlying approach of Filter 3 is described using a typical example. In the following HTML code, the only attribute is `href="http://www.spiegel.de/"`.

```
<a href="http://www.spiegel.de/">Spiegel Web Site</a>
```

Now, length of the anchor text is calculated and saved for each hyperlink (in this example: Spiegel Web Site) into a variable called *length*. Then, the attribute part of the opening tag is substituted with a string of space characters () with a length of (*length* - 7) where the value 7 comes from the length of `<a>`. Therefore, the new hyperlink for this example should be as below:

```
<a <math>length - 7</math>>Spiegel Web Site</a>
```

The above-mentioned explanations of Filter 3 are summarized in Algorithm 3. As can be observed in this algorithm, the `while` loop which is repeated for *n* times calculates the length of the anchor text related to each hyperlink and stores in the *LT* variable. Then, a string of *LT*-7 length is made from the space character and then is inserted into a string variable "Str". Finally, the attribute part of the hyperlink is replaced with the *Str* string.

Algorithm 3: Filter 3.

```
1:  $Hyper = \{h_1, h_2, \dots, h_n\}$ 
2:  $i = 1$ 
3: while  $i \leq n$  do
4:    $length = Length(h_i.anchor\ text)$ 
5:    $String\ Str = new\ String(" ", length - 7)$ 
6:    $substitute(h_i.attributes, Str)$ 
7:    $i = i + 1$ 
8: end while
```

4 DATA SETS AND RESULTS

To evaluate all the three pre-processing algorithms, two suitable data sets are introduced by (Gottron, 2008) and (Mohammadzadeh et al., 2013). Composition and size of the evaluation data sets are given in Tables 1 and 2.

The first dataset contains 2,166 web documents in Arabic, Farsi, Pashto, and Urdu and has been collected from 10 different web sites for evaluation of the main content extraction in right-to-left language web documents. The second corpus contains 9,101 web pages in English, German, and Italian from 12 different web sites and has been established for evaluation of the main content extraction in western language web documents.

Tables 3 and 4 list the obtained results, i.e. recall, precision and F1 score (Gottron, 2007), from combining each of the filters introduced in this paper with the DANAg algorithm. Tables 3 and 4 are again divided into three parts: the first part contains 4 rows and

Table 1: Evaluation corpus of 2,166 web pages.

| web site | size | languages |
|-----------------------------|------|-----------|
| BBC | 598 | Farsi |
| Hamshahri | 375 | Farsi |
| Jame Jam | 136 | Farsi |
| Al Ahram | 188 | Arabic |
| Reuters | 116 | Arabic |
| Embassy of Germany, Iran | 31 | Farsi |
| BBC | 234 | Urdu |
| BBC | 203 | Pashto |
| BBC | 252 | Arabic |
| Wiki | 33 | Farsi |

Table 2: Evaluation corpus of 9,101 web page.

| web site | size | languages |
|-----------|-------|-----------------|
| BBC | 1,000 | English |
| Economist | 250 | English |
| Golem | 1,000 | German |
| Heise | 1,000 | German |
| Manual | 65 | German, English |
| Republica | 1,000 | Italian |
| Slashdot | 364 | English |
| Spiegel | 1,000 | German |
| Telepolis | 1,000 | German |
| Wiki / | 1,000 | English |
| Yahoo | 1,000 | English |
| Zdf | 422 | German |

compares the recalls; whereas the second part compares the precision; and finally, the third section compares the F1 scores. By looking at Tables 3 and 4, one can make the following conclusions:

- As seen in the third part of both Tables 3 and 4, Filter 3 has acquired a better F1 score in comparison with the other two filters in most of the 18 cases. In contrast, Filter 2 has obtained the minimum amount of F1 score as compared to Filters 1 and 3.
- Based on the first part of Tables 3 and 4, it can be observed that Filter 3 has the maximum recall only in 11 web sites out of the total number of 22 web sites, while Filter 3 attains the maximum F1 score in 18 web sites.
- In web sites where the values of recall obtained from Filter 2 or 3 are equal to that of Filter 1, one may judge that the web site does not have any hyperlink in its MC. For example, it can be seen on Economics and ZDF web sites that the recall is equal for all the three filters.
- When Filter 1 has a recall equal to the one in a web site such as Reuters, it can be argued that the web site certainly includes no hyperlink in its MC, thus the other two pre-processors of Filters 2 and

Table 3: Comparison between Recall, Precision and F1 on the corpus in Table 1.

| | | Al-Ahram | BBC Arabic | BBC Pashto | BBC Persian | BBC Urdu | Embassy | Hamshahri | Jame Jam | Reuters | Wikipedia |
|-----------|----------|----------|------------|------------|-------------|----------|---------|-----------|----------|---------|-----------|
| recall | DANAg | 0.942 | 0.990 | 0.959 | 0.997 | 0.999 | 0.949 | 0.993 | 0.963 | 1.0 | 0.613 |
| | Filter 1 | 0.942 | 0.987 | 0.961 | 0.997 | 0.999 | 0.953 | 0.953 | 0.963 | 1.0 | 0.853 |
| | Filter 2 | 0.942 | 0.989 | 0.961 | 0.997 | 0.999 | 0.953 | 0.942 | 0.963 | 1.0 | 0.886 |
| | Filter 3 | 0.942 | 0.987 | 0.959 | 0.997 | 0.999 | 0.949 | 0.993 | 0.97 | 1.0 | 0.81 |
| precision | DANAg | 0.970 | 0.988 | 0.929 | 0.994 | 0.999 | 0.902 | 0.989 | 0.970 | 0.897 | 0.912 |
| | Filter 1 | 0.969 | 0.952 | 0.929 | 0.973 | 0.999 | 0.833 | 0.611 | 0.97 | 0.897 | 0.869 |
| | Filter 2 | 0.969 | 0.691 | 0.918 | 0.961 | 0.999 | 0.831 | 0.498 | 0.97 | 0.897 | 0.852 |
| | Filter 3 | 0.969 | 0.987 | 0.929 | 0.994 | 0.999 | 0.902 | 0.989 | 0.976 | 0.897 | 0.915 |
| F1 | DANAg | 0.949 | 0.986 | 0.944 | 0.995 | 0.999 | 0.917 | 0.991 | 0.966 | 0.945 | 0.699 |
| | Filter 1 | 0.949 | 0.969 | 0.944 | 0.985 | 0.999 | 0.884 | 0.716 | 0.966 | 0.945 | 0.852 |
| | Filter 2 | 0.949 | 0.804 | 0.939 | 0.979 | 0.999 | 0.883 | 0.624 | 0.966 | 0.945 | 0.861 |
| | Filter 3 | 0.949 | 0.985 | 0.944 | 0.996 | 0.999 | 0.917 | 0.991 | 0.973 | 0.945 | 0.852 |

Table 4: Comparison between Recall, Precision and F1 on the corpus in Table 2.

| | | BBC | Economist | Zdf | Golem | Heise | Republica | Spiegel | Telepolis | Yahoo | Wikipedia | Manual | Slashdot |
|-----------|----------|-------|-----------|-------|-------|-------|-----------|---------|-----------|-------|-----------|--------|----------|
| recall | DANAg | 0.893 | 0.966 | 0.963 | 0.997 | 0.945 | 0.997 | 0.942 | 0.979 | 0.955 | 0.578 | 0.680 | 0.318 |
| | Filter 1 | 0.913 | 0.967 | 0.963 | 0.993 | 0.976 | 0.995 | 0.946 | 0.979 | 0.954 | 0.810 | 0.687 | 0.399 |
| | Filter 2 | 0.922 | 0.967 | 0.963 | 0.745 | 0.965 | 0.994 | 0.946 | 0.979 | 0.952 | 0.760 | 0.690 | 0.440 |
| | Filter 3 | 0.890 | 0.967 | 0.963 | 0.999 | 0.964 | 0.996 | 0.941 | 0.980 | 0.953 | 0.787 | 0.686 | 0.372 |
| precision | DANAg | 0.991 | 0.855 | 0.882 | 0.963 | 0.945 | 0.955 | 0.969 | 0.919 | 0.950 | 0.782 | 0.359 | 0.174 |
| | Filter 1 | 0.991 | 0.830 | 0.880 | 0.941 | 0.900 | 0.872 | 0.943 | 0.914 | 0.948 | 0.927 | 0.355 | 0.208 |
| | Filter 2 | 0.935 | 0.732 | 0.812 | 0.707 | 0.830 | 0.792 | 0.938 | 0.914 | 0.944 | 0.882 | 0.356 | 0.192 |
| | Filter 3 | 0.991 | 0.855 | 0.880 | 0.989 | 0.911 | 0.954 | 0.974 | 0.919 | 0.948 | 0.927 | 0.357 | 0.197 |
| F1 | DANAg | 0.924 | 0.990 | 0.912 | 0.979 | 0.955 | 0.970 | 0.949 | 0.932 | 0.952 | 0.645 | 0.401 | 0.209 |
| | Filter 1 | 0.939 | 0.884 | 0.910 | 0.965 | 0.931 | 0.914 | 0.938 | 0.930 | 0.950 | 0.856 | 0.403 | 0.248 |
| | Filter 2 | 0.916 | 0.827 | 0.871 | 0.724 | 0.884 | 0.865 | 0.937 | 0.930 | 0.948 | 0.809 | 0.404 | 0.239 |
| | Filter 3 | 0.922 | 0.900 | 0.910 | 0.994 | 0.931 | 0.970 | 0.951 | 0.932 | 0.950 | 0.840 | 0.404 | 0.236 |

3 have calculated the recall value as one.

- When Filter 2 has a higher recall and a lower precision than the other two filters, it can be concluded that a major part of the extraneous items has been selected as the MC. It is well known that the menus are regarded as one of the additional items in the web pages and each item in the menu is usually built by an anchor tag. Therefore, by application of Filter 2, it would be possible to consider menus as the MC in some of the web sites such as BBC Arabic. However, the value of recall is equal to 0.989 in the web site of BBC Arabic, which is excellent. On the other hand, the value of precision is reported to be 0.804 which is rather poor and indicates existence of some words in the final MC which can hardly be taken as a part of MC.

5 CONCLUSIONS AND FUTURE WORKS

In this paper, three simple pre-processing methods are proposed which can be combined with the line-based main content extraction methods. These methods have been compared with each other and the results show that Filter 3 yields better output values. Especially on hyperlink rich web documents such as Wikipedia, Filter 3 clearly outperforms to the other 2 pre-processing methods.

For the future work, it is recommended to combine the already introduced pre-processing methods with some other state-of-the-art main content extraction approaches, such as CETR (Weninger et al., 2010), Density (Moreno et al., 2009), and ACCB (Gottron, 2008).

ACKNOWLEDGEMENTS

We would like to thank Thomas Gottron for providing us the dataset which was used in the evaluation of this work.

REFERENCES

- Finn, A., Kushmerick, N., and Smyth, B. (2001). Fact or fiction: Content classification for digital libraries. In *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*.
- Gottron, T. (2007). Evaluating content extraction on HTML documents. In *ITA '07: Proceedings of the 2nd International Conference on Internet Technologies and Applications*, pages 123 – 132, Wrexham, Wales, UK.
- Gottron, T. (2008). Content code blurring: A new approach to content extraction. In *DEXA'08: 19th International Workshop on Database and Expert Systems Applications*, pages 29 – 33, Turin, Italy. IEEE Computer Society.
- Kohlschütter, C., Fankhauser, P., and Nejd, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the third ACM international conference on Web search and data mining, WSDM '10*, pages 441–450, New York, NY, USA. ACM.
- Mohammadzadeh, H., Gottron, T., Schweiggert, F., and Nakhaeizadeh, G. (2012). The impact of source code normalization on main content extraction. In *WEBIST'12: 8th International Conference on Web Information Systems and Technologies*, pages 677 – 682, Porto, Portugal. SciTePress.
- Mohammadzadeh, H., Gottron, T., Schweiggert, F., and Nakhaeizadeh, G. (2013). Extracting the main content of web documents based on character encoding and a naive smoothing method. In *Software and Data Technologies, CCIS Series, Springer*, pages 217 – 236. Springer-Verlag Berlin Heidelberg.
- Moreno, J., Deschacht, K., and Moens, M. (2009). Language independent content extraction from web pages. In *Proceeding of the 9th Dutch-Belgian Information Retrieval Workshop*, pages 50 – 55.
- Pinto, D., Branstein, M., Coleman, R., Croft, W. B., King, M., Li, W., and Wei, X. (2002). QuASM: a system for question answering using semi-structured data. In *JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 46 – 55, New York, NY, USA. ACM Press.
- Weninger, T., Hsu, W. H., and Han, J. (2010). CETR: content extraction via tag ratios. In *Proceedings of the 19th International Conference on World Wide Web*, pages 971 – 980. ACM Press.

Detection of Semantic Relationships between Terms with a New Statistical Method

Nesrine Ksentini, Mohamed Tmar and Faïez Gargouri

MIRACL: Multimedia, Information Systems and Advanced Computing Laboratory
University of Sfax, Higher Institute of Computer Science and Multimedia of Sfax, Sfax, Tunisia
ksentini.nesrine@ieee.org, {mohamed.tmar, faiez.gargouri}@isimsf.rnu.tn

Keywords: Semantic Relatedness, Least Square Method, Information Retrieval, Query Expansion.

Abstract: Semantic relatedness between terms plays an important role in many applications, such as information retrieval, in order to disambiguate document content. This latter is generally studied among pairs of terms and is usually presented in a non-linear way. This paper presents a new statistical method for detecting relationships between terms called Least Square Method which defines these relations linear and between a set of terms. The evaluation of the proposed method has led to optimal results with low error rate and meaningful relationships. Experimental results show that the use of these relationships in query expansion process improves the retrieval results.

1 INTRODUCTION

With the increasing volume of textual data on the internet, effective access to semantic information becomes an important problem in information retrieval and other related domains such as natural language processing, Text Entailment and Information Extraction.

Measuring similarity and relatedness between terms in the corpus becomes decisive in order to improve search results (Agirre et al., 2009). Earlier approaches that have been investigating the latter idea can be classified into two main categories: those based on pre-available knowledge (ontology such as wordnet, thesauri, etc) (Agirre et al., 2010) and those inducing statistical methods (Sahami and Heilman, 2006), (Ruiz-Casado et al., 2005).

WordNet is a lexical database developed by linguists in the Cognitive Science Laboratory at Princeton University (Hearst, 1998). Its purpose is to identify, classify and relate in various ways the semantic and lexical content of the English language. WordNet versions for other languages exist, but the English version, however, is the most comprehensive to date. Information in wordnet ;such as nouns, adjectives, verbs and adverbs; is grouped into synonyms sets called synsets. Each group expresses a distinct concept and it is interlinked with lexical and conceptual-semantic relations such as meronymy, hypernymy, causality, etc.

We represent WordNet as a graph $G = (V, E)$ as follows: graph nodes represent WordNet concepts (synsets) and dictionary words; undirected edges represent relations among synsets; and directed edges represent relations between dictionary words and synsets associated to them. Given a pair of words and a graph of related concepts, wordnet computes in the first time the personalized PageRank over WordNet for each word, giving a probability distribution over WordNet synsets. Then, it compares how similar these two probability distributions are by presenting them as vectors and computing the cosine between the vectors (Agirre et al., 2009).

For the second category, many previous studies used search engine collect co-occurrence between terms. In (Turney, 2001), author calculate the Pointwise Mutual Information (PMI) indicator of synonymy between terms by using the number of returned results by a web search engine.

In (Sahami and Heilman, 2006), the authors proposed a new method for calculating semantic similarity. They collected snippets from the returned results by a search engine and presented each of them as a vector. The semantic similarity is calculated as the inner product between the centroids of the vectors.

Another method to calculate the similarity of two words was presented by (Ruiz-Casado et al., 2005) it collected snippets containing the first word from a Web search engine, extracted a context around it, replaced it with the second word and checked if context

is modified in the Web.

However, all these methods measure relatedness between terms in pairs and cannot express them in a linear way. In this paper, we propose a new method which defines linear relations between a set of terms in a corpus based on their weights.

The paper is organized as follows, section 2 is devoted to detailing the proposed method followed by the evaluation in section 3. Finally, section 4 draws the conclusions and outlines future works.

2 PROPOSED METHOD

Our method is based on the extraction of relationships between terms (t_1, t_2, \dots, t_n) in a corpus of documents. Indeed, we try to find a linear relationship that may possibly exist between them with the following form:

$$t_i = f(t_1, t_2, \dots, t_{i-1}, t_{i+1}, \dots, t_n) \quad (1)$$

Least square method (Abdi., 2007), (Miller, 2006) is a frequently used method for solving this kind of problems in an approximate way. It requires some calculus and linear algebra.

In fact, this method seeks to highlight the connection being able to exist between an explained variable (y) and explanatory variables (x). It is a procedure to find the best fit line ($y = ax + b$) to the data given that the pairs (x_i, y_i) are observed for $i \in 1, \dots, n$.

The goal of this method is to find values of a and b that minimize the associated error (Err).

$$Err = \sum_{i=1}^n (y_i - (ax_i + b))^2 \quad (2)$$

Using a matrix form for the n pairs (x_i, y_i) :

$$A = (X^T \times X)^{-1} \times X^T \times Y \quad (3)$$

where A represents vector of values (a_1, a_2, \dots, a_n) and X represents the coordinate matrix of n pairs.

In our case, let term (t_i) the explained variable and the remaining terms of the corpus $(t_1, t_2, \dots, t_{i-1}, t_{i+1}, \dots, t_n)$ the explanatory variables. We are interesting in the linear models; the relation between these variables is done by the following:

$$\begin{aligned} t_i &\approx \alpha_1 t_1 + \alpha_2 t_2 + \dots + \alpha_{i-1} t_{i-1} + \alpha_{i+1} t_{i+1} \\ &+ \dots + \alpha_n t_n + \varepsilon = \sum_{j=1}^{i-1} (\alpha_j t_j) + \sum_{j=i+1}^n (\alpha_j t_j) + \varepsilon \end{aligned} \quad (4)$$

Where α are real coefficients of the model and present the weights of relationships between terms and ε represents the associated error of the relation.

We are looking for a model which enables us to obtain an exact solution for this problem.

Therefore, we proceed to calculate this relation for each document in the collection and define after that the final relationship between these terms in the whole collection. For that, m measurements are made for the explained and the explanatory variables to calculate the appropriate $\alpha_1, \alpha_2, \dots, \alpha_n$ with m represent the number of documents in the collection.

$$\begin{cases} t_i^1 \approx \alpha_1 \cdot t_1^1 + \alpha_2 \cdot t_2^1 + \dots + \alpha_n \cdot t_n^1 \\ t_i^2 \approx \alpha_1 \cdot t_1^2 + \alpha_2 \cdot t_2^2 + \dots + \alpha_n \cdot t_n^2 \\ \vdots \\ t_i^m \approx \alpha_1 \cdot t_1^m + \alpha_2 \cdot t_2^m + \dots + \alpha_n \cdot t_n^m \end{cases} \quad (5)$$

Where t_i^j is the Tf-Idf weight of term i in document j . By using the matrix notations the system becomes:

$$\underbrace{\begin{pmatrix} t_i^1 \\ t_i^2 \\ \vdots \\ t_i^m \end{pmatrix}}_{t_i} \approx \underbrace{\begin{pmatrix} t_1^1 & t_2^1 & \dots & t_n^1 \\ t_1^2 & t_2^2 & \dots & t_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ t_1^m & t_2^m & \dots & t_n^m \end{pmatrix}}_X \times \underbrace{\begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{pmatrix}}_A \quad (6)$$

where X is a TF-IDF (Term Frequency-Inverse Document Frequency) matrix whose rows represent the documents and columns represent the indexing terms (lemmas).

Thus, we seek $A = (\alpha_1, \dots, \alpha_n)$ such as $X \times A$ is more near possible to t_i . Rather than solving this system of equations exactly, least square method tries to reduce the sum of the squares of the residuals. Indeed, it tries to obtain a low associated error (Err) for each relation.

We notice that the concept of distance appears. We expect that $d(X \times A, t_i)$ is minimal, which is written:

$$\min \| X \times A - t_i \| \quad (7)$$

To determine the vector A for each term in a corpus, we applied the least square method on the matrix X for each one.

$\forall i = 1, \dots, n$.

$$A_i = (X^{iT} \times X^i)^{-1} \times X^T[i, \cdot] \times t_i \quad (8)$$

Where X^i is obtained by removing the row of the term i in matrix X and n is the number of terms in a corpus.

$X^T[i, \cdot]$ represents the transpose of the line weight of term i in all documents.

3 EXPERIMENTS

In this paper, we use our method to improve informa-

tion retrieval performance, mainly, by detecting relationships between terms in a corpus of documents.

We focus on the application of the least square method on a corpus of textual data in order to achieve expressive semantic relationships between terms.

In order to check the validity and the performance of our method, an experimental procedure was set up.

The evaluation is then based on a comparison of the list of documents retrieved by a developed information retrieval system and the documents deemed relevant.

To evaluate within a framework of real examples, we have resorted to a textual database, of 1400 documents, called Cranfield collection (Ahrum, 2008)(Sanderson, 2010). This collection of tests includes a set of documents, a set of queries and the list of relevant documents in the collection for each query.

For each document of the collection, we proceed a handling and an analysis in order to lead it to lemmas which will be the index terms. Once the documents are each presented with a bag of words, we have reached by a set of 4300 terms in the whole collection. Hence, matrix X is sized $1400 * 4300$. After that, we applied on it the least square method for each term in order to determine the vector A for each one. The obtained values A_i indicate the relationship between $term_i$ and the remaining terms in the corpus. We obtain another square matrix T with 4300 lines expressing the semantic relationships between terms as follows:

$$\forall i \in 1, 2, \dots, 4300, \forall j \in 1, 2, \dots, 4300$$

$$term_i = \sum (T[i, j].term_j) \quad (9)$$

Example of obtained semantic relationships:

Term *airborn* = 0.279083 *action* + 0.222742 *airforc* + 0.221645 *alon* + 0.259213 *analogu* + 0.278371 *assum* + 0.275861 *attempt* + 0.210211 *behaviour* + 0.317462 *cantilev* + 0.215479 *carrier* + 0.277437 *centr* + 0.216453 *chapman* + 0.22567 *character* + 0.23094 *conecylind* + 0.347057 *connect* + 0.239277 *contact* + 0.225988 *contrari* + 0.217225 *depth* + 0.283544 *drawn* + 0.204302 *eighth* + 0.26399 *ellipsoid* + 0.312026 *fact* + 0.252312 *ferri* + 0.211903 *glauert* + 0.230067 *grasshof* + 0.223152 *histori* + 0.28336 *hovercraft* + 0.380206 *inch* + 0.238555 *inelast* + 0.205513 *intermedi* + 0.275635 *interpret* + 0.235573 *interv* + 0.216454 *ioniz* + 0.319457 *meksyn* + 0.200089 *motion* + 0.223062 *movement* + 0.233753 *multicellular* + 0.376881 *multipli* + 0.436183 *nautic* + 0.219787 *orific* + 0.414204 *proabl* + 0.214005 *propos* + 0.305503 *question* + 0.204316 *read* + 0.222911 *reciproc* + 0.256728 *reson* + 0.237344 *review* + 0.202781 *spanwis* + 0.351152 *telemet* + 0.226465 *ter-*

min + 0.212812 *toroid* + 0.339988 *tunnel* + 0.25228 *uniform* + 0.233854 *upper* + 0.20262 *vapor*.

We notice that obtained relationships between terms are meaningful. Indeed, related terms in a relation talk about the same context, for example the relationship between the lemma *airborn* and the other lemmas (*airborn*, *airforc*, *conecylind*, *action*, *tunnel* ...) talks about the airborne aircraft carrier subject. To test these relationships, we calculate for each one the error rate (Err):

$$Err(term_i) = \sum_{j=1}^{1400} (X[j, i] - (\sum_{k=1}^{i-1} (X[j, k] \times T[i, k]) + \sum_{q=i+1}^{4300} (X[j, q] \times T[i, q])))^2 \quad (10)$$

The obtained values are all closed to zero, for example the error rate of the relationship between term (*account*) and the remaining of terms is $1.5 * 10^{-7}$ and for the term (*capillari*) is $5,23 * 10^{-11}$.

To check if obtained relations improve information retrieval results, we have implemented a vector space information retrieval system which test queries proposed by the Cranfield Collection.

The aim of this kind of system is to retrieve documents that are relevant to the user queries. To achieve this aim, the system attributes a value to each candidate document; then, it rank documents in the reverse order of this value. This value is called the Retrieval Status Value (RSV) (Imafouo and Tannier, 2005) and calculated with four measures (cosines, dice, jaccard and overlap).

Our system presents two kinds of evaluation; firstly, it calculates the similarity (RSV) of a document vector to a query vector. Then, it calculates the similarity of a document vector to an expanded query vector. The expansion is based on the relevant documents retrieved by the first model (Wasilewski, 2011) and the relationships obtained by least square method.

Indeed, if a term of a collection is very related with a term of query ($\alpha \geq 0.5$) and appears in a the relevant returned documents, we add it to a query. Mean Average Precision (MAP) is used to calculate precision of each evaluation. Table1 shows the obtained results.

We notice from this evaluation, that relationships obtained by least square method are meaningful and can provide improvements in the information retrieval process. Indeed, the MAP values are increasing when these relations are used in information retrieval system. For example, our method improves information retrieval results using cosinus measure when $\alpha > 0.6$ with $MAP = 0.21826$ compared to the basic VSM model ($MAP = 0.20858$).

Compare our results with other works, we note

Table 1: Variation of MAP values.

| | VSM | VSM with expanded query | | | |
|---------|---------|-------------------------|----------------|----------------|----------------|
| | | $\alpha > 0.8$ | $\alpha > 0.7$ | $\alpha > 0.6$ | $\alpha > 0.5$ |
| Cosinus | 0.20858 | 0.20654 | 0.21273 | 0.21826 | 0.21822 |
| Dice | 0.20943 | 0.20969 | 0.21529 | 0.21728 | 0.22060 |
| Jaccard | 0.20943 | 0.21043 | 0.21455 | 0.21341 | 0.20642 |
| Overlap | 0.12404 | 0.12073 | 0.12366 | 0.12311 | 0.12237 |

that this new statistical method (least square) improves search results. In (Ahram, 2008), experimental results from cranfield documents collection gave an average precision of 0.1384 which is less than that found in our work (0.21826 with cosine measure, 0.22060 with dice measure).

4 SUMMARY AND FUTURE WORKS

We present in this paper a new method for detecting semantic relationships between terms. The proposed method (least square) defines meaningful relationships in a linear way and between a set of terms using weights of each one which represent the distribution of terms in the corpus.

These relationships give a low error rate. Indeed, they are used in the query expansion process for improving information retrieval results.

As future works, firstly, we will intend to participate in the competition TREC to evaluate our method on a large test collection. Secondly, we will look for how to use these relations in the process of weighting terms and the definition of terms-documents matrix to improve information retrieval results. Finally, we also will investigate these relations to induce the notion of context in the indexing process.

REFERENCES

- Abdi., H. (2007). The method of least squares.
- Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., and Soroa, A. (2009). A study on similarity and relatedness using distributional and wordnet-based approaches. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pages 19–27, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Agirre, E., Cuadros, M., Rigau, G., and Soroa, A. (2010). Exploring knowledge bases for similarity. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).
- Ahram, T. Z. (2008). *Information retrieval performance enhancement using the average standard estimator and the multi-criteria decision weighted set of performance measures*. PhD thesis, University of Central Florida Orlando, Florida.
- Hearst, M. (1998). WordNet: An electronic lexical database and some of its applications. In Fellbaum, C., editor, *Automated Discovery of WordNet Relations*. MIT Press.
- Imafouo, A. and Tannier, X. (2005). Retrieval status values in information retrieval evaluation. In *String Processing and Information Retrieval*, pages 224–227. Springer.
- Miller, S. J. (2006). The method of least squares.
- Ruiz-Casado, M., Alfonseca, E., and Castells, P. (2005). Using context-window overlapping in synonym discovery and ontology extension. In *International Conference on Recent Advances in Natural Language Processing (RANLP 2005)*, Borovets, Bulgaria.
- Sahami, M. and Heilman, T. D. (2006). A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on World Wide Web, WWW '06*, pages 377–386, New York, NY, USA. ACM.
- Sanderson, M. (2010). *Test collection based evaluation of information retrieval systems*. Now Publishers Inc.
- Turney, P. D. (2001). Mining the web for synonyms: Pmi-ir versus lsa on toefl. In *Proceedings of the 12th European Conference on Machine Learning, EMCL '01*, pages 491–502, London, UK, UK. Springer-Verlag.
- Wasilewski, P. (2011). Query expansion by semantic modeling of information needs.

An Approach to Detect Polarity Variation Rules for Sentiment Analysis

Pierluca Sangiorgi^{1,2}, Agnese Augello¹ and Giovanni Pilato¹

¹*ICAR, Istituto di Calcolo e Reti ad Alte Prestazioni, CNR, Consiglio Nazionale delle Ricerche,*

Viale delle Scienze - Edificio 11, 90128, Palermo, Italy

²*INAF, Istituto di Astrofisica Spaziale e Fisica Cosmica - Palermo, via U. La Malfa 153, 90146, Palermo, Italy*

{sangiorgi, augello, pilato}@pa.icar.cnr.it

Keywords: Subjectivity Analysis, Sentiment Analysis, Opinion Mining, Machine Learning.

Abstract: Sentiment Analysis is a discipline that aims at identifying and extract the subjectivity expressed by authors of information sources. Sentiment Analysis can be applied at different level of granularity and each of them still has open issues. In this paper we propose a completely unsupervised approach aimed at inducing a set of words patterns that change the polarity of subjective terms. This is a very important task because, while sentiment lexicons are valid tools that can be used to identify the polarity at word level, working at different level of granularity they are no longer sufficient, because of the various aspects to consider like the context, the use of negations and so on that can change the polarity of subjective terms.

1 INTRODUCTION

In recent years, with the advent of blogs, forums, social communities, product rating platforms and so on, users have become more active in production of large amount of information. Starting from news as well as products reviews and any other information systems that allow on-line user interaction in terms of contents, users provide information through their contributions in the form of opinions, discussions, reviews and so on.

Companies and organizations are increasingly interested in this type of information because it can be used as knowledge resource for operations of market survey, political behavior and, in general, measurement of satisfaction.

Information produced by network users usually regards what is known as their "private states": their opinions, emotions, sentiments, evaluations and beliefs (Quirk et al., 1985) (Banea et al., 2011). The term subjectivity is usually used in literature as a linguistic expression of private state (Banea et al., 2011).

One of the latest and most challenging research task is the automatic detection of subjectivity presence in text, which is named subjectivity analysis. The task also identifying, where possible, its polarity, by classifying it as neutral, positive or negative is defined in literature as sentiment analysis.

Sentiment analysis may be realized at several levels of granularity, like word, sentence, phrase or doc-

ument level. Usually each level of analysis exploits the results obtained by the underlying layers.

What makes these tasks hard to achieve is the ambiguity of words, the context-sensitivity of subjective terms, the need of appropriate linguistic resources for different languages, the presence of negations, the presence of irony, and so on (Montoyo et al., 2012).

In this paper we propose an approach to automatically discover several sequential patterns (i.e. a sequence of tokens in the sentence) in a specific language that change the polarity of subjective words. The novelty of the approach is that it is completely unsupervised, requiring only the use of one single linguistic resource: a sentiment lexicon, which is specific for the language under consideration. This can be done in order to build a polarity variation detector to be used in sentiment analysis applications.

2 RELATED WORKS

In sentiment analysis, the comprehension of sentences polarities is a complex task, which involves the analysis of the composition of the words in the sentences, considering their prior polarities. Several sentiment lexicons (Wilson et al., 2005a), (Stone and Hunt, 1963), (Baccianella et al., 2010), (Strapparava and Valitutti, 2004), can be accessed in order to examine the polarity at a "word-level"; in some cases

these lexicons can be obtained through machine learning approaches (which cluster terms according to their distributional similarity (Turney and Littman, 2003)), or by means of bootstrapping methodologies starting from few term-seeds (Banea et al., 2011)(Pitel and Grefenstette, 2008).

Polarity of words is highly dependent on the domain in which they are used, so that adjectives with positive polarity could have an opposite or a neutral polarity in another domain. Moreover, especially in the context of product reviews users often adopt abbreviations and idioms; therefore methods for the automatic creation of lexicons may be very useful in this context also to improve existing dictionaries.

At a "sentence-level" it is necessary to consider the composition of the word, with their prior polarities, into the phrase. Different compositional models have been proposed in (Yessenalina and Cardie, 2011), (Wu et al., 2011) and (Chardon et al., 2013). In (Hu and Liu, 2004) a set of adjective words (opinion words) is identified using a natural language processing method in order to decide the opinion orientation of a sentence: for each opinion word its semantic orientation is determined, and the opinion orientation is then predicted by analyzing the predominance of positive or negative words. In (Moilanen and Pulman, 2007) a composition model based on a syntactic tree representation has been proposed. Many other factors have to be considered, especially the presence of polarity influencers (Wilson et al., 2005b) (Polanyi and Zaenen, 2006). In (Wilson et al., 2005b) a study on the most important features for recognizing contextual polarity has been performed, and the performance of these features has been evaluated by using several different machine learning algorithms. In (Tan et al., 2012) it has been proposed an automatic approach to detect polarity pattern rules, based on the extraction of typed dependency bi-grams of sentences and the use of Class Sequential Rules (CSRs) to derive polarity class rules from sequential patterns within the bi-grams. In (Tromp and Pechenizkiy, 2013) is proposed an algorithm for polarity detection based on the use of different heuristic rules.

3 AN APPROACH TO DETECT POLARITY VARIATION RULES

The proposed approach is thought to work in an unsupervised manner. It can be run just one time in order to define the rules. A subsequent re-run of this approach is not required, excepted for an extension of the number of rules using other data sets as inputs. The process can be viewed as a black box that pro-

duces, given a set of documents as input, a set of rules in a specific structure that will be described below. The entire process can be decomposed in a sequence of steps as reported in figure 1.

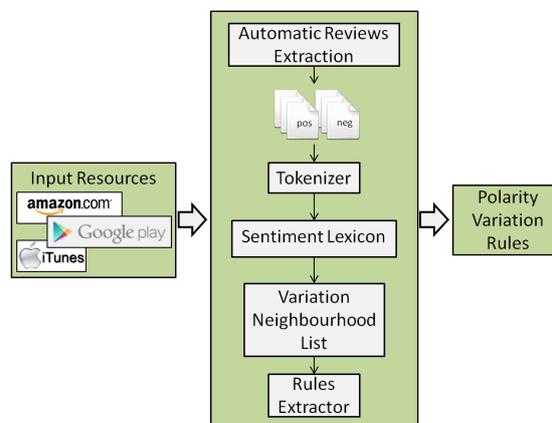


Figure 1: The architecture of the proposed approach.

The main idea is to extract several user reviews expressed in English language from an on-line market store which implements a star rating system. Each one of the reviews represents a short document, usually a sentence, that can be classified as having positive or negative polarity according to the number of stars that are compulsorily inserted by the users in order to publish a review.

Given these two classes of documents, after a simple pre-processing through a tokenizer used for splitting their content into words, we scan each word by exploiting a sentiment lexicon in English language. If in a document with a given polarity is detected a word with a different one, the three words to its left and those to its right are saved in a list.

Repeating this process for all the extracted reviews, we obtain a collection of words typically surrounding subjective words that, "for some reason", are probably responsible for the variation of the polarity of the subjective words.

Without concerning why these rules change words polarity, we can study them in terms of frequency, support and confidence and select the most promising ones.

The collection of these sequential patterns will form a set of polarity variation rules. These rules could be integrated in a polarity variation detector in order to check, during other sentiment analysis applications, if a given sentence matches one of these new rules and therefore a different polarity for the term should be considered.

In a nutshell, the approach is based on four main steps:

1. extraction of a large amount of user reviews from on-line market store with star rating system;
2. sentiment analysis at word level using sentiment lexicons;
3. extraction of the left and right context of a subjective word with inverted polarity respect to that of the sentence;
4. identification of the sequential patterns rules that change the polarity;

It might seem that an approach like this could be replaced by a set of rules manually written by professional linguists, but this is not completely true. This because a linguist can define rules based on the grammar of the language that not always fits the “language” generally used by people on the social media. Furthermore, this approach can be potentially applied to any language of interest, since the polarity variation detection depends only on the tools used.

4 FORMALIZATION OF THE PROPOSED APPROACH

In our approach three main elements can be identified:

1. the input resources which consist of on-line user reviews that we want to analyze;
2. the processing chain;
3. the output which consist of linguistic patterns that cause the change of polarity of subjective terms.

We define reviews as input as an ordered sequence of variable length of words, punctuation and symbols z_i :

$$r_i = \{z_1, z_2, z_3, \dots, z_n\}$$

and the desired output as two lists $P^+ = \{p_i^+\}$ and $P^- = \{p_i^-\}$, respectively for positive and negative sentences, of sequential patterns:

$$p_i = [n_1, n_2, n_3, n_4, n_5, n_6]$$

composed by an experimentally fixed number of terms, some of which empty, that represent the sequence of three terms that are present before (n_1, n_2, n_3) and after (n_4, n_5, n_6) a subjective term, which can cause its polarity variation.

The details of the entire processing chain will be described in the next subsections.

4.1 Acquisition and Data Preparation

The first step of the processing chain accomplishes the task of collecting reviews in order to create the

knowledge resource, rich of subjective terms, from which to extract the polarity variation rules.

To this aim we consider the on-line user reviews as information sources, this because this type of text carries out opinions and sentiments expressed by the users, containing subjectivity terms with an high probability.

In particular, we are considering to retrieve the reviews from the Google Play market store, Amazon, iTunes or other on-line market shop.

This choice is motivated by a fundamental feature characterizing this kind of stores: they implement a star rating system, with scores usually from one to five, that must be inserted by the users in order to publish a review. This feature allows us to extract reviews as classified in terms of opinion (bad or good) expression. We make the supposition that reviews with higher value of stars express positive sentiments of users while reviews with lower value of stars express negative sentiments.

The result of this step is to extract several reviews with their associated stars values for different products and group them by positive or negative expressed opinion in two large class of documents $R^+ = \{r_i^+\}$ and $R^- = \{r_i^-\}$, considering only the stars values.

Once the classes of documents are built, they pass through a tokenizer chain to split the content of reviews in separated terms. The splitting must be done considering space character as well as punctuation, maintaining every remaining alphanumeric terms without filtering.

No other text pre-processing, like stopwords removal, must be done, this because we are interested to all terms that compose the reviews and could be determinant in the rules definition despite they not carry out particular information. For example think about terms like “but”, “not” and so on.

The result of this process is that every review is cleaned from spaces, punctuation and symbols. This lead to a review r_i composed only by terms t_i :

$$r_i = \{t_1, t_2, t_3, \dots, t_n\}$$

4.2 Identification of the Variation Neighborhood Lists

Given the i -th review with positive polarity r_i^+ , composed by a sequence of terms t_j :

$$r_i^+ = \{t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, \dots\}$$

every j -th term t_j is checked through a sentiment lexicon, and whenever the term is detected as being subjective with negative polarity, the term is

marked and its neighbor terms are stored in an array N_i^+ of predetermined size:

$$N_i^+ = [n_{j-3}, n_{j-2}, n_{j-1}, n_{j+1}, n_{j+2}, n_{j+3}]$$

where

$$n = t \text{ (if } t \text{ is a valid terms)}$$

or

$$n = \text{"_"} \text{ (if } t \text{ is empty).}$$

We define as “*Variation Neighborhood List*” a set of sequential patterns of fixed size composed by the terms in the left and in the right context of the subjective terms with discordant polarity, and in this case N_i^+ is an element of the Variation Neighborhood List $L^+ = \{N_i^+\}$ relative to the positive reviews.

This operation is executed for both positive and negative reviews. Generally speaking, we state that whenever a discordance between the prior polarity of a subjective term with the polarity of the review that the term belongs to is detected, a new element of the Variation Neighborhood List is built.

In order to clarify this statement, let us consider, for example, three negative reviews after the tokenizer process:

$r_1^- = \{\text{this, app, is, good, when, it, works, but, ninety, percent, of, the, time, it, will, not, even, open, on, my, phone}\}$

$r_2^- = \{\text{my, previous, review, was, good, as, i, was, satisfied, with, the, app, but, now, feel, that, its, the, worst, app, ever}\}$

$r_3^- = \{\text{not, good, needs, to, be, fixed}\}$

All the three reviews contain the subjective term *good* that have positive prior polarity, which is however discordant with the overall sentiment expressed by the reviews.

In this case we build the three arrays:

$$N_1^- = [\text{this, app, is, when, it, works}]$$

$$N_2^- = [\text{previous, review, was, as, i, was}]$$

$$N_3^- = [\text{"_, _, not, needs, to, be}]$$

Note that the subjective term is discarded and not stored in the array, this because we are interested to its context which can affect its polarity as well as the context of other subjective terms.

The only restriction is to consider only an even number of terms for each context in order to have always patterns with the first half of terms referring to the left context of subjective terms and the second half

to the right.

4.3 Rules Extractor

Once the two Neighborhood Lists L^+ and L^- respectively for positive and negative reviews are built, they are singularly processed in order to extract the two final set of polarity variation rules.

In order to reach this goal, the list is expanded adding all possible combinations of terms of the original sequential patterns extracted in the previews step. Considering the sequential pattern N_2^- of the above example, this means create additional sequential patterns $N_{2,k}$ with $k \in [1, 62]$, like:

$$N_{2,32}^- = [\text{previous, _, _, _, _, _}]$$

$$N_{2,16}^- = [\text{"_, review, _, _, _, _}]$$

$$N_{2,8}^- = [\text{"_, _, was, _, _, _}]$$

$$N_{2,39}^- = \{\text{previous, _, _, as, i, was}\}$$

and so on.

The k -th sequential pattern is built considering the binary encoding of k that leave the original terms in the positions with value equal to 1 and replace the terms with “_” in the positions with value equal to 0. $k=0$ and $k=63$ are not considered because they are respectively the empty pattern and the original pattern. Note that 63 is the maximum number of sequential patterns for each polarity variation detected, but they could be less if the original pattern have empty elements (“_”).

After the expansion process, the next step is to compare all the sequential patterns of the expanded list, remove all the duplicates and saving for each pattern its value of frequency.

Since a pattern with few empty elements is strong and representative of polarity variation, but not probable to occur many times, instead of its frequency we consider a weight value w associated to each pattern; w is defined as:

$$w = (f-1)*l$$

where l is the number of non empty elements of the pattern ($l \in [1, 6]$) and f is the frequency of the pattern in the list.

In this way patterns that occur only once in the list have weight w equal to 0 and therefore they are discarded. Patterns with higher value of l , have an high confidence, but a very low support.

Considering only the frequency does not work: for example, let us consider that a pattern $N_{i,62}$ with a frequency value of 2, implies the existence of 6 patterns with the same frequency but $l=1$ ($N_{i,1}, N_{i,2}, N_{i,4}, N_{i,8}, N_{i,16}, N_{i,32}$), as well as the other “sub-patterns” with all value of l generated from the expansion procedure of the pattern with $l=6$.

Once the weight value w is calculated, for each patterns for the two Neighborhood Lists, using an appropriate value of threshold we remove all the patterns with a value lower than a given threshold .

All the patterns remaining from the cut-off phase represent the polarity variation rules we are looking for and stored as elements p_i in the two list of rules $P^+ = \{p_i^+\}$ and $P^- = \{p_i^-\}$.

They are expressed as vector of 6 terms, someone of which may be empty. These vectors, if used in sentiment analysis tasks, indicate, with a grade of probability, which terms must be present before and/or after a subjective term to cause its polarity variation.

5 CONCLUSIONS AND FUTURE WORK

We think that the work proposed in this paper could be a plausible approach to determine and solve the polarity variation problem in sentiment analysis applications. We are thus interested to develop this approach and we are working in this direction considering Google Play market store as resource for reviews and SentiWordNet, a lexical resource for opinion mining created by manual assignment of polarity to each synset of Wordnet, as sentiment checker. This will let us to define a set of valid rules to be integrated in a computer aided system for real time detection of polarity variation as support for other systems.

The choice to work in this direction, in fact, is not casual and arises from a real necessity occurred during other sentiment analysis works where we are involved. Just for example, during a sentiment analysis process done at word level on a large collection of Google Play market reviews, we have noticed that SentiWordNet found more positive than negative words inside reviews classified as negative with the star rating system. Find any discordant polarity terms inside a sentence it's a typical situation, but not in the numbers we found in our collection. This prompted us to investigate and define a valid approach to solve the problem.

ACKNOWLEDGEMENTS

This work has been partially supported by the POR 2007/2013 - Regione Siciliana - Misura 4.1.1.1. - IDS (Innovative Document Sharing) Research Project and by the PON01_01687 - SINTESYS (Security and INTElligence SYSstem) Research Project.

REFERENCES

- Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).
- Banea, C., Mihalcea, R., and Wiebe, J. (2011). Multilingual Sentiment and Subjectivity Analysis. *ACL 2012*.
- Chardon, B., Benamara, F., Mathieu, Y., Popescu, V., and Asher, N. (2013). Sentiment composition using a parabolic model. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*, pages 47–58, Potsdam, Germany. Association for Computational Linguistics.
- Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '04*, pages 168–177, New York, NY, USA. ACM.
- Moilanen, K. and Pulman, S. (2007). Sentiment composition. In *Proceedings of Recent Advances in Natural Language Processing (RANLP 2007)*, pages 378–382.
- Montoyo, A., Martínez-Barco, P., and Balahur, A. (2012). Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments. *Decision Support Systems*, 53(4):675 – 679.
- Pitel, G. and Grefenstette, G. (2008). Semi automatic building method for a multidimensional affect dictionary for a new language. In *LREC*. European Language Resources Association.
- Polanyi, L. and Zaenen, A. (2006). Contextual valence shifters. In Shanahan, J., Qu, Y., and Wiebe, J., editors, *Computing Attitude and Affect in Text: Theory and Applications*, volume 20 of *The Information Retrieval Series*, pages 1–10. Springer Netherlands.
- Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. (1985). *A comprehensive grammar of the English language*. Longman, London.
- Stone, P. J. and Hunt, E. B. (1963). A computer approach to content analysis: studies using the general inquirer system. In *Proceedings of the May 21-23, 1963, spring joint computer conference, AFIPS '63 (Spring)*, pages 241–256, New York, NY, USA. ACM.
- Strapparava, C. and Valitutti, A. (2004). Wordnet affect: an affective extension of wordnet. In *Proceedings of the 4th International Conference on Language Resources and Evaluation*, pages 1083–1086.
- Tan, L.-W., Na, J.-C., Theng, Y.-L., and Chang, K. (2012). Phrase-level sentiment polarity classification using rule-based typed dependencies and additional complex phrases consideration. *Journal of Computer Science and Technology*, 27(3):650–666.
- Tromp, E. and Pechenizkiy, M. (2013). Rbem: A rule based approach to polarity detection. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining, WISDOM '13*, pages 8:1–8:9, New York, NY, USA. ACM.

- Turney, P. D. and Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Trans. Inf. Syst.*, 21(4):315–346.
- Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. (2005a). Opinionfinder: A system for subjectivity analysis. In *HLT/EMNLP. The Association for Computational Linguistics*.
- Wilson, T., Wiebe, J., and Hoffmann, P. (2005b). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT '05*, pages 347–354, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Wu, Y., Zhang, Q., Huang, X., and Wu, L. (2011). Structural opinion mining for graph-based sentiment representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11*, pages 1332–1341, Stroudsburg, PA, USA. Association for Computational Linguistics.
- Yessenalina, A. and Cardie, C. (2011). Compositional matrix-space models for sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11*, pages 172–182, Stroudsburg, PA, USA. Association for Computational Linguistics.

A General Evaluation Framework for Adaptive Focused Crawlers

Fabio Gasparetti, Alessandro Micarelli and Giuseppe Sansonetti

Department of Engineering, Roma Tre University, Via della Vasca Navale 79, Rome, Italy
{gaspare, micarel, gsansone}@dia.uniroma3.it

Keywords: Adaptive Focused Crawling, Evaluation Framework.

Abstract: Focused crawling is increasingly seen as a solution to increase the freshness and coverage of local repository of documents related to specific topics by selectively traversing paths on the web. The adaptation is a peculiar feature that makes it possible to modify the search strategies according to the particular environment, its alterations and its relationships with the given input parameters during the search. This paper introduces a general evaluation framework for adaptive focused crawlers.

1 INTRODUCTION

Due to the limited bandwidth, storage and resources of traditional computational systems and the rapid growth of the web, focused crawlers aim at building small high-quality and up-to-date repositories of topic-specific pages. Deep analyses of the retrieved pages have also the chance to better address growing dynamic contents, such as news or financial data and promptly alerting about relevant alterations of the retrieved pages.

Pages related to the same topics tend to be neighbours of each other is the fundamental assumption that is often named *topic locality* (Davison, 2000). Thus, the objective of the crawlers is to stay focused, that is, remaining within the neighbourhood in which topic-specific pages have been identified.

In this context, an evaluation methodology is a logical description of the processes and connected elements to be followed to help one better understand a quality evaluation. By following this process, a computer scientist or practitioner can learn what he or she needs to know to determine the level of a performance of a search strategy in a specific context. This paper is geared toward a definition of an evaluation methodology for the adaptive focused crawlers.

At present, focused crawling evaluations that also include adaptivity analysis are not available. One of the reasons could be the difficulty to measure the reaction of crawlers to user needs refinements or alterations of the environment. How long does it take to adapt the crawl to a user relevance feedback and provide new interesting documents? How many environment alterations are tolerable before the crawling

performance falls below a given threshold? Standard methodologies to assess those characteristics, thus allowing comparing different search strategies are yet to be developed.

The aim of this paper is twofold. First, we summarise the different evaluation approaches that have been proposed in the literature, critically discussing the testbed settings and the evaluation metrics. After having identified the most relevant factors to be included in a general framework, we give an account of the fundamental elements for the definition of an evaluation methodology regarding *adaptive* focused crawling systems.

The paper is organized as follows. We first present the most relevant input data that distinguish a specific evaluation in Sections 3.1 and 3.2. Section 3.3 deals with the approaches proposed for the definition of relevance measures of the retrieved pages. In Section 3.4, we consider the resource constraints, while Sect. 3.5 is focused on the measures that better characterise the effectiveness of the search strategies during the crawl. Section 3.6 introduces the evaluation approaches based on comparative analysis, while 3.7 specifically dwells on the assessment of the adaptive behaviour of the crawlers. The following section discusses the related work in the literature. The last section is a conclusion.

2 RELATED WORK

The foremost exploratory research activity on the evaluation of adaptive focused crawlers has been proposed by Menczer *et al.* in (Menczer *et al.*, 2004).

In particular, they compare several crawlers based on machine learning techniques in order to assess both the obtained general performance and some characteristics of adaptivity. The authors' principal goal was to evaluate the benefits of the machine learning versus other approaches. While machine learning has the chance to play a key role in the development of focused crawlers able to automatically adapt their search strategies to the peculiar characteristics of the topics and environment, the proposed framework misses to cover scenarios when the approaches are subjected to continuous updates in the input data (i.e., topics and environment alterations). In this case, adaptivity can be performed either incrementally by continuous update or by retraining using recent batches of data, either new or already visited pages subjected to updates. In this scenario, the relation between the input data and the target variable changes over time.

Several other frameworks have been proposed (Menczer et al., 2001; Chau and Chen, 2003; Srinivasan et al., 2005; Pant and Srinivasan, 2005), but none of them explicitly include adaptive behaviour analysis.

3 AN EVALUATION FRAMEWORK FOR ADAPTIVE FOCUSED CRAWLERS

Defining an evaluation methodology for a *standard* crawler does not require a great effort. Once a subset of the web is available, it is possible to run an instance of the crawl on a workstation and monitor the most important parameters to measure its effectiveness (Cho et al., 1998). The proactivity and autonomy characteristics of the search strategies of focused crawlers, which potentially allow them to explore regions of the web far from the starting points, call for different evaluation approaches.

In addition to that, if the focused crawlers have some sort of adaptivity behaviour w.r.t changes in the environment or the current topics, the evaluation framework should keep track of changes of the performances and behaviour when one of both of these aspects are being altered. Good adaptivity is characterised by changes of unconstructive or disruptive behaviour, often caused by external stimuli, to something more constructive, which is able to fulfil the goal of the search activity.

In the following sections we define the parameters and the most relevant elements that form the evaluation methodology, to be assessed and reported during

the experiments with adaptive focused crawlers.

3.1 Corpus

There are two broad classes of evaluations, system evaluations and user-centred evaluations. The latter measure the user's satisfaction with the system, while the former focuses on how well the system is able to retrieve and rank documents. Several researchers accept that evaluators of adaptive systems should adopt a user-centred evaluation approach because users are both the main source of information and the main target of the application, but manually finding all the relevant documents in a large collections of billion of documents is not practical. User-based evaluation is extremely expensive and difficult to do correctly. A proper designed user-based evaluation must use a sufficiently large, representative sample of potential topics. Such considerations lead researchers to use the less expensive system evaluations.

Of course, technical issues must be addressed in order to construct a collection that is a good sample of the web (Bailey et al., 2003). Nevertheless, "bold" focused crawlers have the chance to take decisions on many different paths and visit pages far from the seed sets, with more chances to end up towards paths of pages not being included in the initial collection. For this reason, standard or predefined collections are rarely employed.

All, or almost all, of the focused crawling evaluations in the literature do not employ any corpus but allow the crawlers to access any document on the web. Web pages continue to change even after they are initially published by their authors and, consequently, it is almost impossible to make comparisons from results obtained by different search strategies, as discussed in Sect. 3.6.

The adaptivity behaviour allows crawlers to dynamically adjust the search strategies to several different and unexpected external alterations. Its evaluation is therefore a complex activity going through the identification and assessment of several variables, sometimes in mutual relationship one another. It is reasonable that a sound evaluation has to consider complex and long-lasting test evaluations to identify those relationships as a function of controlled variations in the input data. A static and large corpus of web documents is the only requirement that guarantees the valid comparison of several outcomes obtained at different times.

3.2 Seeds

A good selection of seed pages guarantees that

enough pages from different communities related to the current topic will be sampled and the crawler exploits the topical locality for finding additional pages in comparison with crawl starting from random seeds. For instance, Daneshpajouh *et al.* (Daneshpajouh *et al.*, 2008) compared various community-based algorithms for discovering good seeds from previously crawled web graphs and discovered that HITS-based ranking is a good approach for this task. Of course, the seed page identification should not be too expensive in terms of computational time. If web corpora are not available, valid sources of seeds may be human-generated directories such as Open Directory Project (ODP)¹, where each category contains links to pages about similar topics.

Of course, seed pages related to the interesting topics make the search for related pages much easier because of the topical locality phenomenon. Srinivasan *et al.* (Srinivasan *et al.*, 2005) provide an interesting mechanism to control the level of difficulty of the crawl tasks by means of the hypertext structure among pages. Once a subset of target pages, that is pages relevant to a topic, is identified, it is possible to collect pages linking to the specified targets by querying one of the online services such as Mozscape². By iterating this *backlink* procedure, it is possible to collect several paths, a subset of them bringing to the target pages. The last pages to be collected are the ones that will be included in the seed set.

The number of iterations I match the level of difficulty of the crawling task. Particularly difficult tasks have a few relevant pages far away from the seed sets. If the crawler is able to find those targets, its edge search strategy has boldness traits favouring the exploration on various different paths. The opposite behaviour of the greedy strategies encourages the exploitation of the current good pages sticking the exploration to their vicinity. An adaptive selection of bold and greedy strategies may rely on the current acquired evidence. For example, once a number of relevant websites have been found, the exploration can be focused on the near linked pages, while bold strategies are valid when no evidence is fruitful and new paths have to be verified. At present, focused crawlers do not explicitly include this form of search strategy adaptivity.

The above-mentioned backlink procedure is the only one that allows the framework to include the recall measure of performance discussed in Sect. 3.5.1. As a matter of fact, the procedure builds up a small corpus of pages, where the good ones are clearly identified.

¹<http://www.dmoz.org>

²<http://moz.com>

3.3 Topic Affinity

Ideal focused crawlers retrieve the highest number of relevant pages while simultaneously traversing the minimal number of irrelevant pages. This leads to significant savings in hardware and network resources, and helps keep the crawl more up-to-date.

One of the first evaluation parameters to take into consideration is the soundness of the retrieved documents' content. The traditional crawlers' goal is to download as many resources as possible, whereas a focused crawler should be able to filter out the documents that are not deemed related to the topics of interest. Focused crawlers respond to the particular information need expressed by topical queries or interest profiles.

Besides monitoring the exploration results, the evaluation of the relatedness of the retrieved documents is also fundamental for the selection of the best routes to follow. For this reason, focused crawlers routinely compute these measures for assigning the priorities to the queued URLs during the exploration. A formal description for the topic of interests is fundamental for effectively driving the crawling to a subset of paths and, of course, it is strictly correlated to the definition of the relatedness measure. Singular domains may also define ad-hoc measures of effectiveness, such as novelty and diversity of page contents (Barbosa and Bangalore,).

In the following sections, we give an account of the most relevant approaches for evaluating the relatedness of the retrieved documents.

3.3.1 Topic Selection

Information searching is the intent that usually motivates the queries driving the focused crawling. Users are willing to locate documents and multimedia concerning a particular topic in order to address an information need or perform a fact-finding or general exploratory activity. These topics can be along a spectrum from very precise to very vague.

A long-lasting research activity aiming at defining a comprehensive classification of user intents for web searching (e.g., (Jansen *et al.*, 2008)) and related IR evaluations (e.g., (Sakai, 2012)) is largely available.

In contrast to search engines, topics submitted to focused crawling are defined by expert users able to accurately select a good representation of their intents. At the same time, those intents can still assume both a broad (e.g., "Find information about Windows 9") or specific scopes (e.g., "Find stocks of DNA sequencing companies").

While automatic approaches to select broad-topic queries are available (see Sect. 3.3.3), specific scope

queries are usually human-generated or extracted from real scenarios (Gasparetti et al., 2014). In spite of that, general evaluation frameworks should take into account both of the typologies in order to assess the benefits of different strategies and adaptivity techniques in the two scenarios.

3.3.2 “Plain” Matching

Several focused crawlers use text similarity measures for comparing the content extracted from the crawled pages and a representation of the topic that drives the search.

If both topics and contents are described by keywords, the relevance between them can be calculated by one of the well-known approaches proposed in the IR, such as:

VSM. Vector Space Model (e.g., (Hersovicia et al., 1998))

NB. Naive Bayes classifiers trained on a subset of documents related to the topic (e.g. (Chakrabarti et al., 1999; Chakrabarti et al., 2002))

SVM. Support Vector Machine (e.g., (Ehrig and Maedche, 2003; Choi et al., 2005; Luong et al., 2009))

NN. Neural networks (e.g., (Menczer and Monge, 1999; Chau and Chen, 2003))

LSI. Latent Semantic Indexing (e.g., (Hao et al., 2011))

The output is usually any real number between 0 and 1:

$$f_m : DxT \rightarrow [0, 1]. \quad (1)$$

where D and T are the representations of the document and topic, respectively.

A comparative evaluation shows how NB classifiers are weak choices for guiding a focused crawler when compared with SVM or NN (Pant and Srinivasan, 2005).

3.3.3 Taxonomy-based Matching

In order to more accurately drive the crawl, some focused crawlers use hierarchical structures for classifying pages (Chakrabarti et al., 1999; Chen et al., 2008). There are several complex hierarchical taxonomies and ontologies available, e.g., Medical Subject Headings, U.S. Patents, ODP and CIDOC Conceptual Reference Model for cultural heritage. Instead of binary classifiers, where each category or class is treated separately, hierarchical categorisation may drop a document into none, one, or more than one category. Users instantiate a crawl by selecting one or more topics in the given taxonomy.

Imagine a hierarchy with two top-level categories, e.g., Computers and Recreation, and several subcategories, such as Computers/Hacking, Computers/Software and Computers/Emulators. In a non-hierarchical model, a word like *computer* is not very discriminating since it is associated with several categories related to computers. In a hierarchical model, more specialized words could be used as features within the top-level Computer category to better choose the right one for a given a document.

Chakrabarti *et al.* (Chakrabarti et al., 1999) determine the relevance of one page analysing its ancestor categories. If one of those ancestors is in the subset of topics selected by the user, the page is further analysed because it covers more detailed topics. The same approach can be employed in an evaluation framework so that relevant documents are not ignored because they do not ideally match the user topic.

Text descriptions of the descendants in the taxonomy can be used to improve the representation of the topic of interests (Chen et al., 2008). Cross-language hierarchical taxonomies can also be employed to allow focused crawlers analyse pages in different languages.

Menczer *et al.* propose to use the ODP taxonomy to automatically generate topics for the evaluations (Menczer et al., 2004). Leaves with five or more links are extracted and used as potential topics. In particular, the text corresponding to the title of the category and the anchors of the external links become a text description of each topic.

Hierarchical categorisation with SVM has been proven to be an efficient and effective technique for the classification of web content (Dumais and Chen, 2000). Other relevant approaches are based on the semantic analysis of the content, e.g., (Limongelli et al., 2011; Gentili et al., 2003; Biancalana et al., 2013).

3.3.4 Predicate-based Matching

A focused crawler estimates the likelihood that each candidate link will lead to further relevant content. Evidence such as links’ anchor text, URL words and source page relevance are typically exploited in estimating link value comparing the text against the current topic of interest.

Aggarawal *et al.* (Aggarwal et al., 2001) propose the definition of arbitrary predicates in order to better perform the resource discovery. Besides simple keywords, predicates may extend to peculiar characteristics of the retrieved pages or properties of the linkage topology. By analysing the characteristics of the collected pages and the values of their predicates, it is possible to understand the statistical relationship between the predicates and the best candidate pages.

For instance, Diligenti *et al.* (Diligenti et al., 2000) use the context-graph idea to learn the characteristics of the best routes examining features collected from paths leading up to the relevant nodes.

Besides sets of keywords, predicates give users the chance to represent the features that the retrieved pages must own in order to be judged relevant. For example, opinion and discourse analysis on contents spread out on a sequence of connected pages might unveil valuable information that strict keyword-based relevance measures on single documents might miss.

While predicates are shown to be fundamental improvements in developing adaptive focused crawlers (Micarelli and Gasparetti, 2007), there are not attempts to use user-defined predicates to evaluate the performance of the crawlers. Predicates are usually very context-dependent, therefore they are strongly affected by the specific goal, situation, domain, task or problem under examination. None of the predicate-based approaches proposed in the literature propose a formal methodology for the definition of those predicates. User-defined predicates are subjective by nature, for this reason they are less suitable for being included in general evaluation frameworks.

3.3.5 Authoritativeness

The overwhelming amount of information on the web related to a potential topic of interest may hinder the ability to make important decisions. One of the advantages of focused crawlers, that is the reduction of the information overload, is only partially achieved when the topics of interest is too general or vague.

Focused crawlers use topic distillation for finding good *hubs*, i.e., pages containing large numbers of links to relevant pages for updating the current queue of URLs to visit (Kleinberg, 1998; Chakrabarti et al., 1999). The purpose of topic distillation is to increase the precision of the crawl, even if there is no trace of the topic keywords in them. Pages and links form a graph structure and connectivity analysis algorithms based on a mutual reinforcement approach is able to extract hubs and authority pages, that is relevant pages pointed by hubs. Different topics may show different topologies of interconnections between web pages. Menczer *et al.* (Menczer et al., 2004) state how iterative algorithms such as HITS and PageRank able to extract meaning from link topology permit the search to adapt to different kinds of topics.

While focused crawlers use hubs for finding new seeds during the crawl, authority measures can be used to evaluate the importance of the retrieved pages. Despite similar performances, different focused crawlers may cover subspaces on the web with low overlap. Due to dissimilar topologies, authority

measures better unveil different outputs and search strategies.

A clear limitation of these measures in an evaluation framework is that they are computed on a partial web graph built by extracting the links from the collected documents obtaining a rough approximation of their values. The use of a static large corpus (e.g., CommonCrawl) can overcome this obstacle.

3.4 Resource Constraints

Focused crawling identifies relevant documents reducing the computational resources required by this task. The principal computational resources are computation time, network bandwidth and memory space. While the processing speed and memory capacity cost unit have been constantly reduced in recent years, network bandwidth poses strong limits on the number of documents that can be downloaded and evaluated.

Focused crawlers based on iterative algorithms such as HITS, e.g., (Cho et al., 1998; Chakrabarti et al., 1999; Rungsawang and Angkawattanawit, 2005) are expected to reduce the rate of page downloads when the set of hypertext documents is large. Most of the current evaluations ignore experiments that extend over 10 thousands of documents and hence they just ignore this issue. Comparative analysis of focused crawlers that include iterative algorithms should clearly state the asymptotic estimates of the complexity, therefore ignoring the efficacy alteration due to potential different implementations of the same algorithms.

In practice, a simple heuristic to determine the CPU usage is monitoring the time elapsed before reaching a given limit of retrieved documents. Results should generally be averaged over several tests and statistical significance values have to be computed in order to reduce the effects of temporary Internet slowdowns and prove the soundness of the evaluation.

3.5 Behaviour Analysis

An evaluation framework of focused crawling strategies has to provide provable guarantees about their performance assessments. However, an algorithm that works well in one practical domain might perform poorly in another. Trend analysis on each topic based on the information accumulated over a period of crawl activity would permit to understand the variations of the performances as a function of explicit or implicit variables. The complexity of the topic, the amount of links in the visited web graph or the unreachable pages are only some of the variables that may strongly alter the behaviour of the focused crawlers. While

an average on several tests may reduce the influence of these variables on the final results, analysing some measures during the crawl gives the chance to get different views on the performance and better characterise the benefits and drawbacks of various strategies in various contexts.

3.5.1 Precision, Recall and Harvest Rate

Precision and recall are two popular performance measures defined in automated IR, well defined for sets. The former P_r corresponds to the fraction of top r ranked documents that are relevant to the query over the total number of retrieved documents, interesting and not.

$$P_r = \frac{\text{found}}{\text{found} + \text{false alarm}} \quad (2)$$

while recall R_r is the proportion of the total number of relevant documents retrieved in the top r (cutoff) over the total number of relevant documents available in the environment:

$$R_r = \frac{\text{found}}{\text{found} + \text{miss}} \quad (3)$$

Precision and recall are virtually independent by the relatedness measure definition, therefore it is possible to employ one of the above-mentioned measures in order to identify relevant and irrelevant documents.

As pointed out in (Chakrabarti et al., 1999), the recall indicator is hard to measure because it is impossible to clearly derive the total number of documents relevant to a topic due to the vastness of the web, unless the backlink procedure for the seed selection discussed in Sect. 3.2 is chosen.

If the precision of the fetched pages is computed during the crawl, the curve of *harvest rate* $h_r(n)$ for different time slices of the crawl is obtained, where n is the current number of fetched pages (Chakrabarti et al., 1999). This measure indicates if the crawler gets lost during the search or if it is able to constantly keep the search over the relevant documents. The harvest rate becomes a critical measures for analysing the behaviour of the crawlers after alterations of the environment or topics of interests (see Sect.3.7).

3.5.2 Deep Web Strategies

On a different note, most search engines cover what is referred to as the publicly indexable Web but a large portion of the Internet is dynamically generated and such content typically requires users to have prior authorisation, fill out forms, or register (Raghavan and Garcia-Molina, 2001). Other information refers to Twitter or Facebook posts, links buried many layers

down in a structured website, or results that sit so far down the standard search results that typical users will never find them. This covert side of the Internet is commonly referred to as the hidden/deep/invisible web. Hidden web content often corresponds to precious information stored in specialised databases. Focused crawlers have the chance to include novel deep web crawling strategies in order to find out additional relevant documents (Zheng et al., 2013). A feasible measure to assess the effectiveness of these strategies is based on the comparison of the retrieved pages with the collection of pages retrieved by popular search engines. A large subset of relevant documents that does not overlap with the search engines' collections is expected for good deep web strategies.

While these strategies are golden features useful in several contexts and, therefore, required to be evaluated during the crawl, a very few attempts have been proposed, and all of them limit the scope of their techniques to strategies for specific portions of websites (Bergholz and Chidlovskii, 2003; Liakos and Ntoulas, 2012).

3.6 Comparative Analyses

Section 3.1 discussed how focused crawling can be seen as a particular instance of the IR task, which goal is selecting a subset of documents from a large collection relevant to a given topic. For this reason, the long-lasting research activity in the IR evaluation has the chance to support new frameworks for focused crawlers.

Several IR experiments are designed following the Cranfield paradigm, where same sets of documents, topics and measures are used for various approaches that are considered in isolation, freed as far as possible from the contamination of operational variables. The experimental design calls for same corpus of hypertext documents and same topics, with the computation of the same effectiveness measures in order to directly compare different approaches' outcomes. A performance comparison between adaptive, non-adaptive and unfocused crawlers (e.g., breadth-first, random strategy) can be easily obtained.

While hypertext test collections have been often used in the IR domain (e.g., the ones provided by the Text Retrieval Conference TREC), they show several drawbacks in the focused crawling as discussed in Par.3.1.

Current focused crawling evaluations (e.g., (Srinivasan et al., 2005; Menczer et al., 2004)) follow a hybrid approach, where each round of tests are based on the same topic and measures but each single strategy is evaluated allowing the crawler to directly access the

web. The authors make the assumption that the web is not being altered between two evaluations. Except if the evaluations take place very quickly one after another, this assumption is clearly wrong.

A partial workaround consists in caching the accessed pages so that future requests and evaluations for that data can be served ignoring potential occurred alterations. Due to the locality reference of crawlers, it is also possible to cache pages that are connected by the ones that have been retrieved during the crawl (i.e., prefetching). While caching can simulate similar testbeds between evaluations of different search strategies, it fails to maintain consistency between the cache's intermediate storage and the location where the data resides. For example, home pages of news websites such as CNN.com are usually altered several times a day while other sections are not. Caching techniques that store only part of these websites cannot reproduce a valid image of their hypertext structure and reachable content. Once again, a large snapshot of the web is the only feasible way to guarantee the same platform for experimentation for various search strategies.

3.7 Adaptivity

Menczer *et al.* associate *adaptivity* to the approaches that include any sort of machine learning techniques for guiding search (Menczer *et al.*, 2004). Adaptive techniques are basically seen as means to better understand the environment and its peculiar relationships with the topic. The environment and the topics are perceived as static features.

On a different note, Micarelli and Gasparetti (Micarelli and Gasparetti, 2007) extend the definition of adaptive focused crawlers to the ones able to address potential variations in the environment or in the topic definition. As a matter of fact, two relevant adaptive crawlers (Menczer and Monge, 1999; Gasparetti and Micarelli, 2003) show both adaptive behaviour implementing multi-agent evolutionary or optimisation algorithms. For this reason, we should like to propose a methodology and measures to effectively assess this form of adaptation, whatever technology is chosen for the implementation of the focused crawlers.

3.7.1 Domain Adaptivity

Empirical analysis of web page changes combined with estimates of the size of the web states how an amount close to 5% of the indexable web must be subjected to re-index daily by search engines to keep the collection up-to-date (Brewington and Cybenko, 2000). Several statistical approaches aim at predicting whether a page will change based on the change

frequency for the same page observed over some past historical window (Radinsky and Bennett, 2013). Nevertheless to our knowledge, there is not any focused crawler that implements a scheduling policy for revisiting web pages and adaptively alters its search strategy accordingly. Tight restrictions on the network bandwidth do not allow to allocate enough resources for collecting evidence about change rates of pages. Without these data, robust computation of temporal change patterns and prediction activity are not possible.

Singular exceptions are focused crawlers based on genetic or ant paradigm approaches (Gasparetti and Micarelli, 2003; Menczer *et al.*, 2004). In both the approaches, a population of autonomous agents are able to visit the environment collecting evidence about potential alterations of content and hypertext structure. In spite of that, the authors have not included the domain adaptivity characteristic in the evaluation of their approaches, nor have not future comparative studies done.

At the same time, estimating the importance of each page during the discovery of the web graph is one of the goals of the focused crawlers (Abiteboul *et al.*, 2003). In some circumstances, such as stock market news, the importance is affected by the freshness of the published content. Efficient focused crawling strategies call for a better understanding of this relationship and adaptively change the behaviour of search for uncover the largest number of important resources.

A feasible approach for evaluating these aspects is by empirically measuring the time requested to revisit a carefully defined set of pages that have been subjected of alterations in their content. In particular, once a large set of cached pages have been collected by previous evaluations, it is possible to identify the subset W_a of these pages that have been altered since the beginning of the tests. By randomly choosing a subset $W'_a \subseteq W_a$ that satisfies a given percentage of unique changes (e.g., 5%) and, at the same time, is related to the current topics, new evaluations are performed. Good adaptive strategies will access to these pages W'_a sooner keeping the available computational resources the same (see Par. 3.4).

3.7.2 Topic Adaptivity

A more subtle form of adaptivity regards the topic that guides the crawl. Queries or interest profiles might be altered during the crawl in various ways:

- Generalisation: A similar or new topic seeking more general information than the previous one;
- Specialisation: A similar or new topic seeking

more specific information than the previous one;

- Reformulation: A new topic that can be viewed as neither a generalisation nor a specialisation, but a reformulation of the topic.

While traditional IR approaches consider each query independently one another, focused crawlers have the chance to exploit collected evidence during previous crawls to drive future exploratory activities on similar topics saving computational resources.

Generalisation and specialisation are two forms of topic alterations that can be easily automated by employing a taxonomy-based representation of topics, as discussed in Sect. 3.3.3. Lower levels of these forms of topic organisations correspond to specialisation while upper levels to generalisation. During comparative analysis, several search strategies may be affected by the same topic alteration. By monitoring the impact of this alteration on the performance measures (e.g., average topic affinity of pages) it is possible to identify the approaches that better exploit the collected evidence being able to promptly adapt the exploration.

4 CONCLUSIONS

The major contribution of the present position paper is to propose an extended evaluation framework for focused crawlers able to take into consideration adaptivity behaviours. A developed discussion on the limitations of the current approaches allowed us to identify relevant features that have currently been ignored in the literature. Moreover, using the lessons learned from the previous crawler evaluation studies, the proposed framework makes explicit reference to the measures proven to be fundamental so far.

We are currently planning to apply the described methodology in a real scenario, where a comparative analysis will analyse the performance of the most popular adaptive focused crawlers.

REFERENCES

- Abiteboul, S., Preda, M., and Cobena, G. (2003). Adaptive on-line page importance computation. In *Proceedings of the 12th International Conference on World Wide Web, WWW '03*, pages 280–290, New York, NY, USA. ACM.
- Aggarwal, C. C., Al-Garawi, F., and Yu, P. S. (2001). Intelligent crawling on the world wide web with arbitrary predicates. In *Proceedings of the 10th International Conference on World Wide Web, WWW '01*, pages 96–105, New York, NY, USA. ACM.
- Bailey, P., Craswell, N., and Hawking, D. (2003). Engineering a multi-purpose test collection for web retrieval experiments. *Inf. Process. Manage.*, 39(6):853–871.
- Barbosa, L. and Bangalore, S. In Macdonald, C., Ounis, I., and Ruthven, I., editors, *CIKM*, pages 755–764. ACM.
- Bergholz, A. and Chidlovskii, B. (2003). Crawling for domain-specific hidden web resources. In *Proceedings of the Fourth International Conference on Web Information Systems Engineering, WISE '03*, pages 125–, Washington, DC, USA. IEEE Computer Society.
- Biancalana, C., Gasparetti, F., Micarelli, A., and Sansonetti, G. (2013). Social semantic query expansion. *ACM Trans. Intell. Syst. Technol.*, 4(4):60:1–60:43.
- Brewington, B. E. and Cybenko, G. (2000). How dynamic is the web? In *Proceedings of the 9th International World Wide Web Conference on Computer Networks : The International Journal of Computer and Telecommunications Networking*, pages 257–276, Amsterdam, The Netherlands, The Netherlands. North-Holland Publishing Co.
- Chakrabarti, S., Punera, K., and Subramanyam, M. (2002). Accelerated focused crawling through online relevance feedback. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 148–159, New York, NY, USA. ACM Press.
- Chakrabarti, S., van den Berg, M., and Dom, B. (1999). Focused crawling: A new approach to topic-specific web resource discovery. In *Proceedings of the 8th World Wide Web Conference (WWW8)*, pages 1623–1640, Toronto, Canada.
- Chau, M. and Chen, H. (2003). Comparison of three vertical search spiders. *Computer*, 36(5):56–62.
- Chen, Z., Ma, J., Han, X., and Zhang, D. (2008). An effective relevance prediction algorithm based on hierarchical taxonomy for focused crawling. In Li, H., Liu, T., Ma, W.-Y., Sakai, T., Wong, K.-F., and Zhou, G., editors, *Information Retrieval Technology*, volume 4993 of *Lecture Notes in Computer Science*, pages 613–619. Springer Berlin Heidelberg.
- Cho, J., Garcia-Molina, H., and Page, L. (1998). Efficient crawling through url ordering. *Computer Networks and ISDN Systems*, 30(1–7):161–172.
- Choi, Y., Kim, K., and Kang, M. (2005). A focused crawling for the web resource discovery using a modified proximal support vector machines. In Gervasi, O., Gavrilova, M., Kumar, V., Lagan, A., Lee, H., Mun, Y., Taniar, D., and Tan, C., editors, *Computational Science and Its Applications ICCSA 2005*, volume 3480 of *Lecture Notes in Computer Science*, pages 186–194. Springer Berlin Heidelberg.
- Daneshpajouh, S., Nasiri, M. M., and Ghodsi, M. (2008). A fast community based algorithm for generating web crawler seeds set. In Cordeiro, J., Filipe, J., and Hammoudi, S., editors, *WEBIST (2)*, pages 98–105. INSTICC Press.
- Davison, B. D. (2000). Topical locality in the web. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 272–279, New York, NY, USA. ACM Press.

- Diligenti, M., Coetzee, F., Lawrence, S., Giles, C. L., and Gori, M. (2000). Focused crawling using context graphs. In *VLDB '00: Proceedings of the 26th International Conference on Very Large Data Bases*, pages 527–534, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Dumais, S. and Chen, H. (2000). Hierarchical classification of web content. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '00*, pages 256–263, New York, NY, USA. ACM.
- Ehrig, M. and Maedche, A. (2003). Ontology-focused crawling of web documents. In *SAC '03: Proceedings of the 2003 ACM symposium on Applied computing*, pages 1174–1178, New York, NY, USA. ACM Press.
- Gasparetti, F. and Micarelli, A. (2003). Adaptive web search based on a colony of cooperative distributed agents. In Klusch, M., Ossowski, S., Omicini, A., and Laamanen, H., editors, *Cooperative Information Agents*, volume 2782, pages 168–183. Springer-Verlag.
- Gasparetti, F., Micarelli, A., and Sansonetti, G. (2014). Exploiting web browsing activities for user needs identification. In *International Conference on Computational Science and Computational Intelligence (CSCI 2014)*. IEEE Computer Society Conference Publishing Services.
- Gentili, G., Micarelli, A., and Sciarrone, F. (2003). Infoweb: An adaptive information filtering system for the cultural heritage domain. *Applied Artificial Intelligence*, 17(8-9):715–744.
- Hao, H.-W., Mu, C.-X., Yin, X.-C., Li, S., and Wang, Z.-B. (2011). An improved topic relevance algorithm for focused crawling. In *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on*, pages 850–855.
- Hersovicia, M., Jacovia, M., Maareka, Y. S., Pellegb, D., Shtalhaima, M., and Ura, S. (1998). The shark-search algorithm an application: tailored web site mapping. In *Proceedings of the 7th World Wide Web Conference(WWW7)*, Brisbane, Australia.
- Jansen, B. J., Booth, D. L., and Spink, A. (2008). Determining the informational, navigational, and transactional intent of web queries. *Inf. Process. Manage.*, 44(3):1251–1266.
- Kleinberg, J. (1998). Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th annual ACM-SIAM symposium on Discrete algorithms*, pages 668–677, San Francisco, CA, USA.
- Liakos, P. and Ntoulas, A. (2012). Topic-sensitive hidden-web crawling. In *Proceedings of the 13th International Conference on Web Information Systems Engineering, WISE'12*, pages 538–551, Berlin, Heidelberg. Springer-Verlag.
- Limongelli, C., Sciarrone, F., and Vaste, G. (2011). Personalized e-learning in moodle: The moodle-ls system. *Journal of E-Learning and Knowledge Society*, 7(1):49–58.
- Luong, H. P., Gauch, S., and Wang, Q. (2009). Ontology-based focused crawling. In *Information, Process, and Knowledge Management, 2009. eKNOW '09. International Conference on*, pages 123–128.
- Menczer, F. and Monge, A. E. (1999). Scalable web search by adaptive online agents: An infospiders case study. In Klusch, M., editor, *Intelligent Information Agents*, pages 323–340. Springer-Verlag, Berlin, Germany.
- Menczer, F., Pant, G., and Srinivasan, P. (2004). Topical web crawlers: Evaluating adaptive algorithms. *ACM Trans. Internet Technol.*, 4(4):378–419.
- Menczer, F., Pant, G., Srinivasan, P., and Ruiz, M. E. (2001). Evaluating topic-driven web crawlers. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01*, pages 241–249, New York, NY, USA. ACM.
- Micarelli, A. and Gasparetti, F. (2007). Adaptive focused crawling. In Brusilovsky, P., Kobsa, A., and Nejdl, W., editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 231–262. Springer Berlin Heidelberg.
- Pant, G. and Srinivasan, P. (2005). Learning to crawl: Comparing classification schemes. *ACM Trans. Inf. Syst.*, 23(4):430–462.
- Radinsky, K. and Bennett, P. N. (2013). Predicting content change on the web. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, WSDM '13*, pages 415–424, New York, NY, USA. ACM.
- Raghavan, S. and Garcia-Molina, H. (2001). Crawling the hidden web. In *VLDB '01: Proceedings of the 27th International Conference on Very Large Data Bases*, pages 129–138, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Rungsawang, A. and Angkawattawit, N. (2005). Learnable topic-specific web crawler. *J. Netw. Comput. Appl.*, 28(2):97–114.
- Sakai, T. (2012). Evaluation with informational and navigational intents. In *Proceedings of the 21st International Conference on World Wide Web, WWW '12*, pages 499–508, New York, NY, USA. ACM.
- Srinivasan, P., Menczer, F., and Pant, G. (2005). A general evaluation framework for topical crawlers. *Information Retrieval*, 8(3):417–447.
- Zheng, Q., Wu, Z., Cheng, X., Jiang, L., and Liu, J. (2013). Learning to crawl deep web. *Inf. Syst.*, 38(6):801–819.

A Domestic Ecosystem Driven by a Networked Intelligence

Luca Ferrari, Matteo Gioia, Gian Luca Galliani and Bruno Apolloni

Department of Computer Science, University of Milano, Milan, Italy
{ferrari, gioia, galliani, apolloni}@di.unimi.it

Keywords: Advanced Domestic, Internet of Things, MQTT Protocols, Learning Algorithms.

Abstract: We describe a diffuse control system for household appliances rooted in an Internet of Things network empowered by a cognitive system. The key idea is that these appliances constitute an ecosystem populated by a plenty of devices with common features, yet called to satisfy in an almost repetitive way needs that may be very diversified, depending on the user preferences. This calls for a network putting them in connection and a cognitive system that is capable to interpret the user requests and translate them into instructions to be transmitted to the appliances. This in turn requires a proper architecture and efficient protocols for connecting the appliances to the network, as well as robust algorithms that concretely challenge cognitive and connectionist theories to produce the instructions ruling the appliances. We discuss both aspects from a design perspective and exhibit a mockup where connections and algorithms are implemented.

1 INTRODUCTION

Control theory, as an offspring of cybernetics, is deeply rooted in the concept of feedback (Wiener, 1948) since the forties of the previous century. In the same period of time a different notion of control in terms of self adapting systems emerged with the first modern hypotheses of the brain computing facilities within the connectionist framework (Rosenblatt, 1958). These two ways of controlling a dynamic system ran in parallel up until the nineties, when hybrid control systems began exploiting synergies by conventional controls and by either recurrent neural networks (Ku and Lee, 1995) or reinforcement learning algorithms (Sutton and Barto, 1998). The Internet of Things paradigm suggests a renewed synergy between the two approaches. Namely, the network connecting things on the Web enjoys many properties of a neural network in terms of both connectivity and elementariness of the messages normally exchanged between the devices and their huge number. However, rather than distributed as in the connectionist paradigm, computations are expected to be more efficient if they are performed in a centralized way, yet exploiting the capillarity of the information the network may bring them – hence we call it *networked intelligence*. *Per se*, the machine where computations are done is immaterial, so that we may assume them to be carried out (and possibly distributed too) everywhere in the cloud. However, as with the multilayer perceptron (Haykin, 1994), the information manage-

ment is dealt with better through a hierarchical architecture than through a distributed one.

This is the idea we pursue in our ecosystem. Namely, we are devising a system constituted by an ensemble of household appliances ruled by a social network which is from time to time committed by a user to operate them optimally with respect to a given task. For instance, the user asks the network to have trousers perfectly washed by his washing machine. In reply, the network sends directly to the machine a sequence of instructions, call it *recipe*, such as “charge water, heat water to 35 degrees”, etc., which drive the machine to carry out a perfect washing. On the one hand this a typical procedure we are used to expect from our personal devices. On the other one, the way of implementing the procedure is rather hard. In an extreme synthesis we need:

1. electronics allowing the network to communicate with the machine, possibly overriding its micro-controller logic;
2. a logic able to produce recipes that are optimal in respect to many criteria, from user preferences all the way to ecological goals such as water or electricity saving. The logic must learn how to reach these objectives from the feedback coming both from devices and from users. Hence it is a cognitive system.
3. an architecture and protocols vehiculating signals between devices and the networked intelligence in a safe and efficient way.

The good news is that the proposed ecosystem has some appeal, so that it got funded by the European Community with a horizon of 30 months (project Social&Smart (*SandS*)-<http://www.sands-project.eu/>). In this paper we report some of the project's progress as to both the architecture-and-protocols and logics. In addition, we describe an early mockup where instruction and signal dispatchings are enabled by proper circuits. Specifically, in Section 2 we introduce the architecture as the backbone of the project. In Section 3 we briefly discuss protocols, while in Section 4 we show the mockup in detail. In the last section we conclude the paper by saying where we are now and what we expect at the end of the project.

2 SandS ARCHITECTURE

Diffuse control may be viewed as an evolution of WEB 2.0 in terms of a social network whose goal is the dispatching of (optimally controlled) activities rather than the providing of information services. This fits well both with the paradigm of Internet of Things, as for architecture, and with the modern reword expectation from the interaction between members within an evolved society. In this new framework a member is spared from doing boring (because repetitive) activities and is enabled to enjoy better ways of life thanks to the automatic contribution of other members. The supporting infrastructure is a community of personal appliances realized through their connection in Internet. Let us explore these aspects in depth.

2.1 Social Network

Social networks can be seen as a repository of information and knowledge that can be queried when needed to solve problems or to learn procedures. The following definition has been proposed by Vannoy and Palvia [1] in the study of social models for technology adoption of social computing: "an ensemble of intra-group social and business actions practiced through group consensus, group cooperation, and group authority, where such actions are made possible through the mediation of information technologies, and where group interaction causes members to conform and influences others to join the group". We intend to update this definition through the novel idea of building social computing systems where part of the computations is hidden to the social network player, thus representing a form of subconscious computing.

Namely, in our social network we will distinguish between conscious and subconscious computing. The former is defined by the decisions and actions performed by the players (i.e. the users) on the basis of the information provided by the social service. The latter is realized by the data processing performed automatically and autonomously by the web service in order to search for or produce the information offered to the social players, or to fulfill other purposes, such as data mining for the advertising industry. It is an intelligent subconscious computing when new solutions to new or old problems are generated on demand.

We instantiate this new social network in the SandS project in the realm of household appliances and domestic services. The term *eahouker* – meaning easy household worker – is introduced in this context to denote the household appliance user empowered by the social network and social intelligence. In an extreme synthesis the project deals with a social network aimed at producing recipes with tools of computational intelligence, to be dispatched to household appliances grouped in the homes through a domestic wifi network. A *recipe* is a set of scheduled, possibly conditional, instructions (hence a sequence of parameters such as water temperature or soak duration) which completely define the running of an appliance. They are managed by a home middleware – called domestic infrastructure (DI) – in order to be properly transmitted to the appliance through suitable protocols. The entire contrivance is devised to optimally carry out ordinary housekeeping tasks through a proper function of house appliances with a minimal intervention on the part of the user. Feedbacks are sent by users and appliances themselves to the network intelligence to close the permanent recipe optimization loop, with offline tips and advices on the part of the appliance manufacturers. An electronic board will interface each single appliance to the DI (see Fig. 1 (Apolloni et al., 2013))

2.2 The Architecture

This architecture has head in the cloud and feet on the appliances. The general scheme is the following (See Fig. 2):

- user and appliances located in a house;
- both of them are interfaced to Internet through a home router: the latter as machines endowed with some transmission device, the former comfortably sitting on a recliner, sending short orders to DI from time to time;
- many houses refer to the same web-services provider. The connection of the router to the

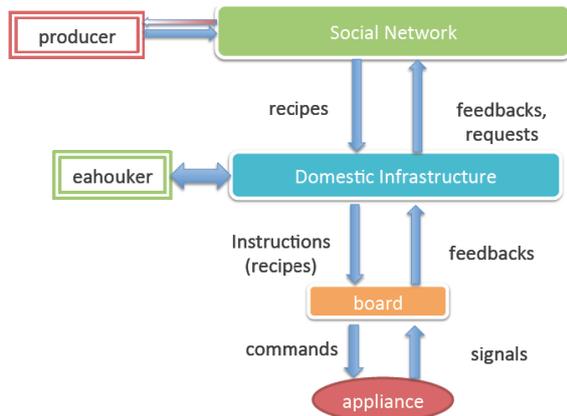


Figure 1: A SandS project synopsis. *eahouker* is a combination of the words easierly and houseworker.

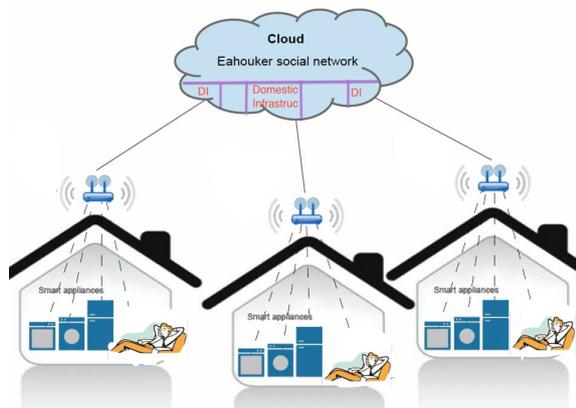


Figure 2: SandS domotic system.

provider is per usual, the internal networking of the communication is committed to a special protocol and a concentrator board if necessary;

- services are delivered in the frame of a social network.

With respect to this consolidated scenario, the peculiarities of our system are:

1. We look for a system as uninvasive as possible. This entails that extra-hardware must be reduced to the minimum; hence: the appliances to be as usual supplied by the stores plus the interfacing chip (Arduino board in the provisional solution we propose); no need for having a computer on, no set-top box.
2. We look for a system as undemanding as possible. This calls for plug&play procedures when an appliance is installed in or removed from the home. Analogously, the need for user input of data must be kept to a minimum; likewise the basic commands for activating the appliances. The

way of inputting data is immaterial; for instance by smartphone, tablet, pc or other specific gadget.

3. we look for extremely adaptive services/recipes. This passes through two functional blocks: a social network which issues recipes on demand to the single user and a DI which administers the recipe according to rules on a home-by-home basis.
4. We look for a system that is transparent to the user. Hence, by definition, the DI is on the cloud, as is the social network. Transparency binds the entire chain from user to social network and back.
5. We look for an intelligent system. Here the intelligence of social network members is involved first when a service is initialized and then regularly when the user sends fuzzy feedbacks. While this is insufficient for creating a true intelligent system, we can fill up this task with a series of algorithms on a proper eahouker database (EDB) that constitute the Networked Intelligence (NI) in the core of the social network that is assigned to issuing recipes.

2.3 A Functional Layout

Fig. 3 gives an intuitive representation of the interactions between the system elements (Grana et al., 2013). The SandS Social Network mediates the interaction between a population of users (the eahoukers – hence ESN the name of the social network), each one with his/her own set of appliances. ESN has a repository of tasks that have been posed by the eahoukers and a repository of recipes for the use of appliances. These two repositories are related by a map between (to and from) tasks and recipes. This map does not need to be one-to-one. Blue dashed arrows correspond to the path followed by the eahouker queries, which are used to interrogate the database of known/solved tasks. If the task is already known, then the corresponding recipe can be returned to the eahouker appliance (solid black arrows). The eahouker can express his/her satisfaction with the results (dashed blue arrows). When the queried task is unknown and unsolved then the social network will request a solution from the SandS Networked Intelligence that will consists in a new recipe deduced from past knowledge stored in the recipe repository. This new solution will be generated by intelligent system reasoning. The repository of recipes solving specific tasks can be loaded by the eahoukers while commenting among themselves on specific cases, or by the appliance manufacturing companies as an example of use to foster sales by offering customer additional appealing services. These situations correspond to the

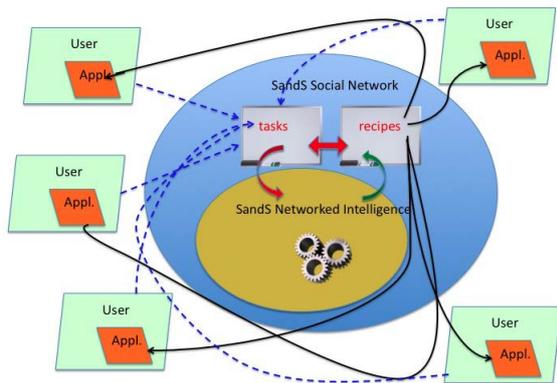


Figure 3: Social and Smart system functional layout.

conscious computing done on the social web service by human agents. The role of the Networked Intelligence is to provide the subconscious computing that generates innovation without eahouker involvement.

2.4 A Conceptual Map of a SandS Session

Figure 4 contains the conceptual map graph (Graña et al., 2013). The main elements are highlighted in red: the eahouker and the appliance, in fact all SandS, is designed to mediate between them. Green boxes contain explicitly active computational modules such as the natural language processing module, the task and recipe managers, the networked intelligence and the domestic middleware. The blue circle highlights the instrumental key of the system: the appliance recipe. The magenta box denotes a hidden reinforcement learning module which the eahouker is not aware of. The SandS session is started by the eahouker stating a task in natural language. The natural language processing module analyzes this expression obtaining a task description which is suitable for a formal search in databases. The task manager explores a task database looking for the best match to the proposed task. If there is an exact match life will be easy for the recipe manager which needs only to retrieve the corresponding recipe. In general, the task manager will select a collection of best matching task descriptions to be presented (or not) to the eahouker to assess the accuracy of the interpretation of his/her intentions by the system. The eahouker may agree to the best matching task descriptions. The recipe manager reads them and continues to explore the recipe database looking for best matches, or proceeds to ask the networked intelligence for the enrichment of the recipe database with new solutions that may better fulfill the task posed by the user. The recipe manager produces a selection of recipes and a best matching



Figure 4: Description of a SandS session by a conceptual map.

recipe. For the engaged user, the selection of recipes may allow him/her to either ponder them and influence the recipe choice or simply trust on the first manager selection by default. This may even be an additional source of feedback to the networked intelligence.

When the recipe is selected, it is downloaded to the appliance via the domestic middleware, which controls its execution. This includes any communication with the user to operate the appliance (i.e. opening the appliance door). The domestic middleware produces a monitoring followup of appliance function that may be shown to the user to keep him/her informed of progress, expected time to completion, etc. The appliance produces a final result, which is returned to the eahouker. Then the eahouker expresses his/her satisfaction, which is the main feedback for all processes.

3 THE TELECOMMUNICATION INFRASTRUCTURE

Appliances and DI are permanently connected in order to send each other status information and commands. A persistent connection is the best solution for an event oriented project like this. Events can be triggered by either DI side or by the appliance side and information can be sent immediately using the already established connection.

Information is secured by cryptographic functions and encapsulated in MQTT (<http://mqtt.org/>) frames at application layer. The connections, ciphering and MQTT encapsulation are managed by a connection manager module on the DI as shown on the figure 5.

Communication between DI and appliances should be encrypted to prevent that an attacker can get information or control the appliance. The wireless

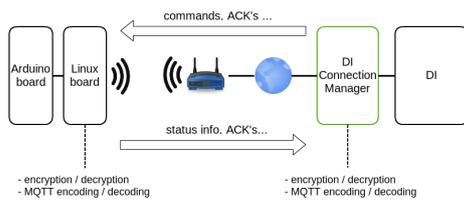


Figure 5: Information exchange between DI and appliance.

network of the user would be secured with a WPA2 system, and so an attack from someone not authenticated on the network will be prevented. However, a possible attack could come from someone who has illegitimately accessed and authenticated on the network. To prevent this and also to prevent possible attacks on the channel between the users router and the DI an additional security layer is introduced (see Fig. 6) by adding a Transport Layer Security (TLS—<http://www.ietf.org/rfc/rfc2246.txt>) to the communication loop.

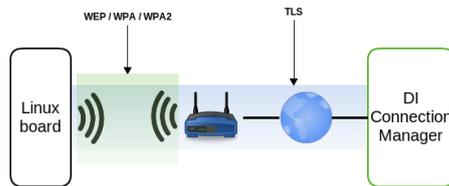


Figure 6: TLS and Wifi encryption.

TLS also allows the use of a certificate to confirm that DI is really DI (and not an attacker supplanting the DI). Even encrypting the communication an attacker could just resend some previously recorded frames in order to send orders to the appliance.

4 AN ENABLING MOCKUP

We have set up a first mockup in the Computer Science Department labs of University of Milano. From a purely operational perspective, it consists of two - three white goods wifi connected to Internet, each with its own Arduino board, in order to execute recipes and send back status signals. For the moment only a washing machine (*wm*) is being tested (see Fig. 7). We use an Arduino MEGA ADK board, an Arduino WIFI SHIELD (see Fig. 8) and a Sitecom wifi router connected to the university network. On this hardware we implement a communication protocol solving, albeit in a preliminary way, problems both of security and of exact message addressing from middleware to appliance and back, within a MQTT-like service (MQTTstandard,) that is sketched in a next section. The protocol is event



Figure 7: A early SandS mockup.

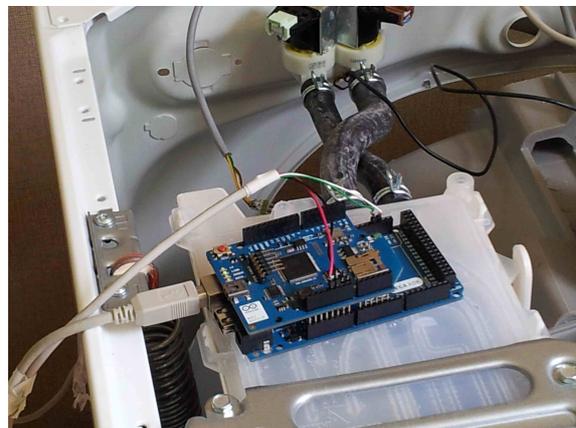


Figure 8: The Arduino interface.

driven, so messages are sent only if state variables change value. The appliance detection is realized in plug&play mode. The management of the appliance, in terms of specification and recipe requests, is done via a browser on any mobile or fixed device. The appliance is located on the ground floor of the Computer Science Department, while the console is on the third floor of the same building. A distance of around 300 m is covered by ethernet wiring, while the last 10 m (including two concrete walls) are gapped via wifi connection. A sketch of the parameters managed by the browser is reported in fig. 9. In the following we propose early considerations on the various parts of the mockup.

4.1 Computational Requirements to the Interfacing Board

Quite simply, we ask to the Arduino board to act as a transducer: in input an instruction to the *wm* actuators, in output the corresponding signal to be trans-

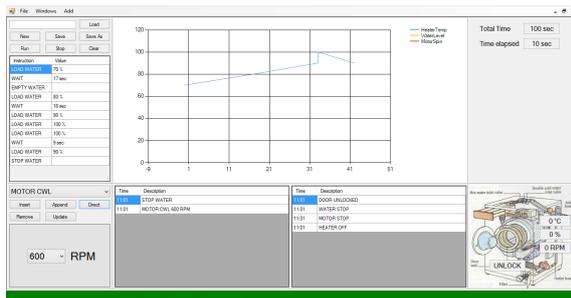


Figure 9: A screenshot of the washing machine monitor.

mitted to the wm microcontroller. From this perspective, the signals overwrite the native microcontroller firmware with a set of commands that change over time. Thus the microcontroller receives one command at a time consisting of the actuator ID and the value of the parameter to be fixed, where the time is decided in principle by the DI. On the one hand this sort of unitary code (one parameter per actuator) requires common tricks to manage multiparameter actuators. On the other hand, we distinguish two time scales, i.e. a time granularity triggering the shift between scales. The general philosophy is: a sequence of instructions, each lasting more than the time granularity, is dispatched at the proper time by the DI. Vice-versa, sequences of instructions lasting less than the time granularity are encoded into a single instruction that is interpreted and sequenced directly by the Arduino board. This requires a finite automaton to reside in the board, both to carry out the above sequencing and to manage a series of security checks to avoid plant damages and logical inconsistencies. As a matter of fact, even one shot instructions such as “heat the water” requires a set of controls – e.g. the water level to prevent the system from burning out in case of an empty drum – which are managed locally by Arduino. However, as to granularity, this instruction falls in the first category. One example of the second category is the alternating running of the wm motor during the pre-wash phase.

This is the basic situation with regard to normal running. In addition, the board is called upon to supervise an anomalous running at two levels – warning and alarm – to which different standby or shutdown sequences may follow as a consequence of the recognized drawback.

Thus the interfacing board is required for computational power to store the instructions to be decoded locally and to manage local time during the implementation of these instructions. For these purposes, the Arduino capacities (RAM 256 KB, processor AT-MEGA2570) prove to be sufficient.

4.2 Communication Requirements to the Interfacing Board

The communication bandwidth necessary for the mockup is normally very low. It increases relatively when the appliance is woken-up by a recipe execution request. The communication is stroked by the *keep alive* message sent by Arduino to the DI. Thus, every 5 seconds, the board waits for instructions by the DI. In the current debugging phase the board sends the appliance status (water temperature and level, spin rpm) as a more informative payload. In a greater detail, the message from board to DI is structured as follows, as a further simplification of MQTT scheme:

```
[CMD,EVN,VAN,VALUE,CHK]
[ = start of message
CMD = command.
EVN = (progressive) Event number.
VAN = Var number (indexing the boiling up variables).
VALUE = Var value (type:Byte, Int, Long o String).
CHK = Checksum23
] = end of message
```

Conversely, the DI sends instructions to the board. We index each instruction with a sequential number within a recipe ID. This indexing may prove suitable for both debugging and appliance quality control purposes. The message format is analogous, with internal event number mating with the one of the sent event for a correct reckoning of the queues.

```
[CMD,EVN,VAN,VALUE,CHK]
[ = start of message
CMD = command.
EVN = Event number (the same of the message which is the answer to).
VAN = Var number.
VALUE = Var value.
CHK = Checksum
] = end of message
```

Once the connection is stated between DI and board, it proceeds in full duplex mode. The connection is open and maintained by the appliance through the *keep alive* messages. In Fig. 10 we can see a segment of the conversation between appliance and DI, as collected by the Arduino debugger.

The time granularity of the recipe transmission takes into account various idle times. Besides normal messages related to the recipe execution, both communication addressees may send overriding messages concerning alarm status and various kinds of shutdown/standby commands.

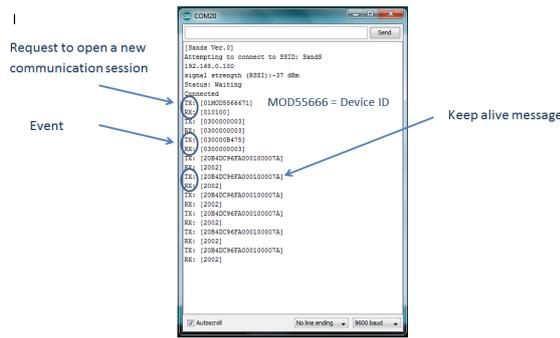


Figure 10: A screenshot of the Arduino debugger.

4.3 Degrees of Freedom in Overwriting the Microcontroller Software

In principle the Arduino Interface is an open-hardware/open software device, so that any person may install it on his/her white good without requiring any license. However, its implementation requires a set of weld junctions and software for overwriting the appliance microcontroller that are specific to the appliance in hand. A further step in the mockup implementation will be to extend the software standardization to a maximum. But overwriting the microcontroller is not an easy task. Rather, we may expect the emergence of small business third parties – for instance, free lance professionals or white good stores with modern selling strategies – which are specialized in this task. For instance, Fig. 11 reports a typical calibration curve relating the water temperature level in the drum and the electrical signal transmitted to the board. Nevertheless, the obvious expectation is that appliances’ manufacturers will pursue their own business interests and support this new paradigm of appliance usage by embedding their products with the necessary electronics.

In this early implementation, we played the role of networked intelligence substitute by featuring a few recipes by ourselves. This gave us the opportunity to appreciate the degrees of freedom of this operation and the optimality criteria we may pursue as a counterpart. Namely we jointly considered the following goals: 1) washing efficacy, 2) energy consumption, 3) water consumption and 4) environmental pollution.

5 CONCLUSIONS

While the project is still at an initial stage, we have come to see the high value it offers in terms both of user convenience and of environmental advantage. For instance, questions re the benefit and ecological

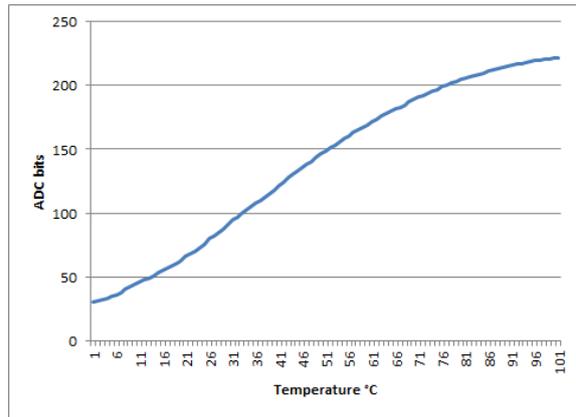


Figure 11: A calibration curve.

profitability of using a bio soap product or reducing the amount of wash water become theoretical opinions no longer. They may be experimented by the single user and many similar users as well within the eahouker social network, so that these questions may find a concretely *scientific* answer.

Though the concrete operations necessary for realizing the mockup delineate a scenario where each single user may implement her/his SandS terminal on home appliances with some technical help, the main way to implement our paradigm passes through a both moral and economical suasion to convince the manufacturers that social appliances are more efficient and rewarding. To achieve this goal SandS project will realize the above discussed architecture in order to set up an initial social network of eahoukers where the benefits of the paradigm will be tossed in concrete by a thousand members. These benefits and the members enjoying them will be the authentic promoters of the SandS ecosystem, again all on the spirit of exploiting actual facts besides sentences.

REFERENCES

Apolloni, B., Fiasch, M., Galliani, G., Zizzo, C., Caridakis, G., Siolas, G., Kollias, S. D., Romay, M. G., Barrieto, F., and Jose, S. S. (2013). Social things - the sands instantiation. In *WOWMOM*, pages 1–6. IEEE.

Grana, M., Apolloni, B., Fiasche, M., Galliani, G., Zizzo, C., Caridakis, G., and et al. (2013). Social and smart: towards an instance of subconscious social intelligence. In *1st Workshop on Innovative European Policies and Applied Measures for Developing Smart Cities, 14th EANN*.

Graña, M., Nuñez-Gonzalez, J. D., and Apolloni, B. (2013). A discussion on trust requirements for a social network of eahoukers. In *HAIIS*, pages 540–547.

Haykin, S. (1994). *Neural Networks: A Comprehensive*

- Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition.
- Ku, C.-C. and Lee, K. Y. (1995). Diagonal recurrent neural networks for dynamic systems control. *IEEE Trans. Neural Netw. Learning Syst.*, 6(1):144–156.
- MQTTstandard. <http://mqtt.org/>.
- Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408.
- Sutton, R. S. and Barto, A. G. (1998). *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition.
- Wiener, N. (1948). *Cybernetics, Or Control and Communication in the Animal and the Machine*. Wiley, New York.

SPECIAL SESSION ON BUSINESS APPS

FULL PAPERS

Towards Process-driven Mobile Data Collection Applications

Requirements, Challenges, Lessons Learned

Johannes Schobel, Marc Schickler, Rüdiger Pryss, Fabian Maier and Manfred Reichert
Institute of Databases and Information Systems, University of Ulm, James-Franck-Ring, Ulm, Germany
{johannes.schobel, marc.schickler, ruediger.pryss, fabian.maier, manfred.reichert}@uni-ulm.de

Keywords: Process-aware Information System, Electronic Questionnaire, Mobile Business Application.

Abstract: In application domains like healthcare, psychology and e-learning, data collection is based on specifically tailored *paper & pencil* questionnaires. Usually, such a paper-based data collection is accomplished by a massive workload regarding the processing, analysis, and evaluation of the data collected. To relieve domain experts from these manual tasks and to increase the efficiency of the data collection process, we developed a generic approach for realizing process-driven smart mobile device applications based on process management technology. According to this approach, the logic of a questionnaire is described in terms of an explicit process model whose enactment is driven by a generic process engine. Our goal is to demonstrate that such a process-aware design of mobile business applications is useful with respect to mobile data collection. Hence, we developed a generic architecture comprising the main components of mobile data collection applications. Furthermore, we used these components for developing mobile electronic questionnaires for psychological studies. The paper presents the challenges identified in this context and discusses the lessons learned. Overall, process management technology offers promising perspectives for developing mobile business applications at a high level of abstraction.

1 INTRODUCTION

Recently, smart mobile applications have been increasingly used in a business context. Examples include simple applications (e.g., task management), but also sophisticated analytic business applications. In particular, smart mobile devices can be used for enabling flexible mobile data collection as well (Pryss et al., 2013). Thereby, data can be collected with sensors (e.g., pulse sensor), communicating with the smart mobile device (Schobel et al., 2013), or with smart form-based applications (Pryss et al., 2012). Examples of applications requiring such a mobile data collection include clinical trials, psychological studies, and quality management surveys.

Developing a mobile data collection application requires specific knowledge on how to implement smart mobile applications. Furthermore, it requires domain-specific knowledge, usually not available to the programmers of these applications. Hence, to avoid a gap between business needs and IT solutions, continuous and costly communication between domain and IT experts becomes necessary. To improve this situation, a framework for rapidly developing and evolving mobile data collection applications is indis-

pensable. In particular, respective business applications should be easy to maintain for non-computer (i.e., domain) experts as well. Our overall vision is to enable domain experts to develop mobile data collection applications at a high level of abstraction. Specifically, this paper focuses on the process-driven design, implementation, and enactment of mobile questionnaire applications, which support domain experts with their daily data collection tasks.

As application domain for demonstrating the benefits of our approach we choose psychological studies. Here, domain experts mostly use paper-based questionnaires for collecting required data from subjects. However, such a paper-based data collection shows several drawbacks, e.g., regarding the structure and layout of a questionnaire (e.g., questions may still be answered, even if they are no longer relevant or needed), or the later analysis of the answers (e.g., errors might occur when transferring the collected paper-based data to electronic worksheets).

To cope with these issues and to understand the subtle differences between paper-based and electronic questionnaires in a mobile context, first of all, we implemented several questionnaire applications for smart mobile devices and applied them in real and

sophisticated application settings (Liebrecht, 2012; Schindler, 2013). In particular, we were able to demonstrate that electronic questionnaires relieve domain experts from costly manual tasks, like the transfer, transformation and analysis of the collected data. As a major drawback, the first applications we had implemented were hard-coded and required considerable communication with domain experts. As a consequence, these applications were neither easy to maintain nor extensible. However, in order to avoid a gap between the domain-specific design of a questionnaire and its technical implementation enacted on smart mobile devices, an easy to handle, flexible and generic *questionnaire system* is indispensable.

From the insights we gained during the practical use of the above mentioned mobile applications as well as from lessons learned when implementing other kinds of mobile applications (Robecke et al., 2011), we elicited the requirements for electronic questionnaire applications that enable a flexible mobile data collection. In order to evaluate whether the use of process management technology contributes to the satisfaction of these requirements, we mapped the logic of a complex questionnaire from psychology to a process model, which was deployed to a process engine. It then served as basis for driving the execution of questionnaire instances at realtime. In particular, this mapping allows us to overcome many of the problems known from paper-based questionnaires. In turn, the use of a process modeling component as well as a process execution engine in the given context, raised additional challenges, e.g., related to the process-driven execution of electronic questionnaires on and the mobile data collection with smart mobile devices. The implemented questionnaire runs on a mobile device and communicates with a remote process engine to enact psychological questionnaires. As a major lesson, we learned that process management technology may not only be applied in the context of business process automation, but also provides a promising approach for generating mobile data collection applications. In particular, a process-driven approach enables non-computer experts to develop electronic questionnaires for smart mobile devices as well as to deploy them on respective devices in order to collect data with them.

In detail, the contributions of this paper are as follows:

- We discuss fundamental problems of paper-based questionnaires and present requirements regarding their transfer to smart mobile devices.
- We provide a mental model for mapping questionnaires to process models. Further, we illustrate this mental model through a real-world applica-

tion scenario from the psychological domain.

- We present a generic architecture for applications running on smart mobile devices that can be used to model, visualize and enact electronic questionnaires. This approach relies on the provided mental model and uses process models to define and control the flow logic of a questionnaire.
- We share fundamental insights we gathered during the process of implementing and evaluating the mobile data collection application.

The remainder of this paper is structured as follows: Section 2 discusses issues related to paper-based questionnaires. Further, it elicits the requirements that emerge when transferring a paper-based questionnaire to an electronic version running on smart mobile devices. Section 3 describes the mental model we suggest for meeting these requirements. In Section 4, we present the basic architecture of our approach for developing mobile data collection applications. Section 5 provides a detailed discussion, while Section 6 presents related work. Finally, Section 7 concludes the paper with a summary and outlook.

2 CASE STUDY

In a case study with 10 domain experts, we analyzed more than 15 paper-based questionnaires from different domains, particularly questionnaires used in the context of psychological studies. Our goal was to understand the issues that emerge when transferring paper-based questionnaires to smart mobile devices. Section 2.1 discusses fundamental issues related to *paper & pencil* questionnaires. Then, Section 2.2 elicits fundamental requirements for their electronic mobile support.

2.1 Paper-based Questionnaires

We analyzed 15 paper-based questionnaires from psychology and medicine. In this context, a variety of issues emerged. First, in the considered domains, a questionnaire must be *valid*. This means that it should have already been applied in several studies, and statistical evaluations have proven that the results obtained from the collected data are representative. In addition, the questions are usually presented in a neutral way in order to not affect or influence the subject (e.g., patient). Creating a valid instrument is one of the main goals when setting up a psychological questionnaire. In particular, reproducible and conclusive

results must be guaranteed. Furthermore, a questionnaire may be used in two different modes. In the *interview mode*, the subject is interviewed by a supervisor who also fills in the questionnaire; i.e., the supervisor controls which questions he is going to ask or skip. This mode usually requires a lot of experience since the interviewer must also deal with questions that might be critical for the subject. The other mode we consider is *self-rating*. In this mode, the questionnaire is handed out to the subject who then answers the respective questions herself; i.e., no supervision is provided in this mode

Another challenging issue of paper-based questionnaires concerns the *analysis* of the data collected. Gathered answers need to be transferred to electronic worksheets, which constitutes a time-consuming and error-prone task. In particular, note that during the interviews or the self-filling of a questionnaire, typographical errors or wrong interpretations of given answers might occur. In general, both sources of error (i.e., errors occurring during the interviews and errors occurring during the transcription) decrease the quality of the data collected, which further underlines the need of an electronic support for flexible and mobile data collection.

In numerous interviews we conducted with 10 domain experts from psychology, additional issues have emerged. Psychological studies are often performed in developing countries, e.g., surveying of child soldiers in rural areas in Africa (Crombach et al., 2013; Liebrecht, 2012). *Political restrictions* regarding data collection further require attention and influence the way in which interviews and assessments may be performed by domain experts (i.e., psychologists). Since in many geographic regions the available infrastructure is not well developed, data collected with paper-based questionnaires is usually digitalized in the home country of the scientists responsible for the study. Taking these issues into account, it is not surprising that psychological studies last from *several weeks up to several months*. From a practical point of view, this raises the problem of allocating enough space in luggage to transfer the paper-based questionnaires safely to the home country of the respective researcher.

Apart from these *logistic problems*, we revealed issues related to the interview procedure itself. In particular, it has turned out that questionnaires must often be *adapted to a particular application context* (e.g., changing the language of a questionnaire or adding / deleting selected questions). Such adaptations (by authorized domain experts) must be propagated to all other interviewers and smart mobile devices respectively in order to keep the results valid and compara-

ble.

Considering these issues, we had additional discussions with domain experts from psychology, which revealed several requirements discussed in the next section.

2.2 Requirements

In the following, we discuss basic requirements for the mobile support of electronic questionnaires. We derived these requirements in the context of case studies, literature analyses, expert interviews, and hands-on experiences regarding the implementation of mobile data collection applications (Crombach et al., 2013; Ruf-Leuschner et al., 2013; Isele et al., 2013). Especially, when interviewing domain experts, fundamental requirements could be elicited. The same applies to the various paper & pencil questionnaires we analyzed.

The major requirements are as follows:

- R1 (Mobility).** The process of collecting data should be highly flexible and usually requires extensive interactions. Data may have to be collected even though no PC is available at the place the questionnaire should be filled in. For example, consider data collection at the bedside of a patient in a hospital or interviews conducted by psychologists in a meeting room. PCs are often disturbing in such situations, particularly if the interviewer is “hiding” himself behind a screen. To enable flexible data collection, the device needs to be portable instead. Further, it should not distract the participating actors in communicating and interacting with each other.
- R2 (Multi-User Support).** Since different users may interact with a mobile questionnaire, multi-user support is crucial. In addition, it must be possible to distinguish between different user roles (e.g., interviewers and subjects) involved in the processing of an electronic questionnaire. Finally, a particular user may possess different roles. For example, an actor could be interviewer in the context of a specific questionnaire, but subject in the context of another one.
- R3 (Support of Different Questionnaire Modes).** Generally, a questionnaire may be used in two different modes: interview and self-rating mode (cf. Section 2.1). These two modes of questioning diverge in the way the questions are posed, the possible answers that may be given, the order in which the questions are answered, and the additional features provided (e.g., freetext notes). In general, mobile electronic questionnaire applications should allow for both modes. Note, that

this requirement is correlated with R2 as the considered roles determine the modes available for a questionnaire.

R4 (Multi-Language Support). The contents of a questionnaire (e.g., questions and field labels) may have to be displayed in different languages (e.g., when conducting a psychological study globally in different countries). The actor accessing the questionnaire should be allowed to choose among several languages.

R5 (Skeuomorphism and Paper-based Style). To foster the comprehensibility of an electronic questionnaire and to ensure its validity, the latter should be designed in the same style (i.e., same order and structure) as the corresponding paper-based version. For example, this means, that the structuring of a questionnaire in different pages must be kept.

R6 (Native Application Style). Any mobile support of electronic questionnaire application must consider different mobile operating systems (e.g., Android or iOS). In this context, standard control elements of the respective mobile operating system should be used to ensure familiarity of users with the elements of the questionnaire when running the latter on their preferred smart mobile device.

R7 (Self-Explaining User Interface). The user interface should be easy to understand and provide intuitive interaction facilities. Furthermore, users should be guided through the process of collecting data with their smart mobile devices.

R8 (Maintainability). Questionnaires evolve over time and hence may have to be changed occasionally. Therefore, it should be possible to quickly and easily change the structure and content of an electronic questionnaire; e.g., to add a question, to edit the text of a question, to delete a question, or to change the order of questions. In particular, no programming skills should be required in this context; i.e., domain experts (e.g., psychologists) should be able to introduce respective changes at a high level of abstraction.

Especially, requirement R8 constitutes a major challenge, which necessitates a high level of abstraction when defining and changing electronic questionnaires, which may then be enacted on a variety of smart mobile devices. To cope with this challenge, we designed a specific mental model for electronic questionnaires, which will be presented in Section 3.

3 MENTAL MODEL

To transfer paper-based questionnaires into electronic ones and to meet the requirements discussed, we designed a mental model for the support of mobile electronic questionnaires (cf. Figure 1). According to this model, the logic of a paper-based questionnaire is described in terms of a process model, which is then deployed to a process management system. The latter allows creating and executing process (i.e., questionnaire) instances.

Generally, a process model serves as template for specifying and automating well defined processes based on process management technology. In addition, adaptive process management systems allow for dynamic process changes of instances to handle unplanned exceptional situations as well (Reichert and Weber, 2012). In the following, we show that process-awareness is useful for realizing applications other than business process automation as well. Applying the process paradigm and process management technology in the context of mobile data collection, however, raises additional challenges (cf. Section 5). We will show how to realize a process-aware approach that guides users in filling in electronic questionnaires based on process management technology.

3.1 Process Model and Instances

As opposed to traditional information systems, process-aware information systems (PAIS) separate process logic from application code. This is accomplished based on process models, which provide the schemes for executing the respective processes (Weber et al., 2011). In addition, a process model allows for a visual (i.e., graph-based) representation of the corresponding process, comprising activities (i.e., process steps) as well as the relations (i.e., control and data flow) between them. For control flow modeling, both control edges and gateways (e.g., ANDsplit, XORsplit) are provided.

A process model P is represented as a directed, structured graph, which consists of a set of nodes N (of different types NT) and directed edges E (of different types ET) between them. We assume that a process model has exactly one start node ($NT = StartFlow$) and one end node ($NT = EndFlow$). Further, a process model must be connected; i.e., each node n can be reached from the start node. In turn, from any node n of a process model, the end node can be reached. In this paper, we solely consider block-structured process models. Each branching (e.g. parallel or alternative branching) has exactly one entry and one exit node. Further, such blocks may be

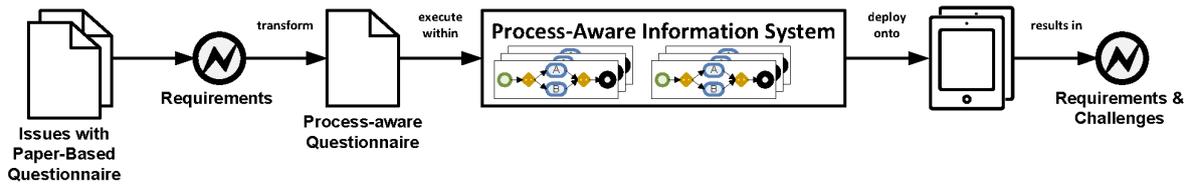


Figure 1: Mental model.

nested, but are not allowed to overlap (Reichert and Dadam, 2009). In turn, data elements D correspond to global variables, which are connected with activities through data flow edges ($ET_DataFlow$). These data elements can either be read ($ReadAccess$) or written ($WriteAccess$) by an activity (Reichert and Dadam, 1998) from the process model. Figure 3 shows an example of such a process model.

In turn, a process instance I represents a concrete case that is executed based on a process model P . In general, multiple instances of a process model may be created and then concurrently executed. Thereby, the state of an instance is defined by the marking of its nodes and edges as well as the values of its data elements. Altogether, respective information corresponds to the execution history of an instance. The process engine has a set of execution rules which describe the conditions under which a node may be activated (Reichert and Dadam, 1998). If its end node is reached, a process instance terminates. An example of how to map a questionnaire to a process model is provided in Section 3.2.

3.2 Mapping a Questionnaire to a Process Model

Our mental model enabling a process-driven enactment of questionnaires is as follows: We define both the contents and the logic of a questionnaire in terms of a process model. Thereby, pages of the questionnaire logically correspond to process activities, whereas the flow between these activities specifies the logic of the questionnaire. The questions themselves are mapped to process data elements, which are connected with the respective activity. There are separate elements containing the text of a question, which can be read by the activity. Moreover, there are data elements that can be written by the activity. The latter are used to store the given answers for a specific question. Figure 2 gives an overview of the mapping of the elements of a questionnaire to the ones of a process model.

To illustrate the process-driven modeling of electronic questionnaires, we present a scenario from psychology. Consider the process-centric questionnaire model from Figure 3. Its process logic is described in

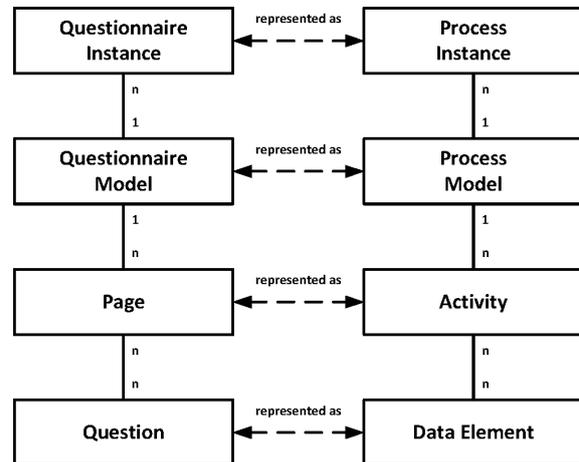


Figure 2: Mapping a Questionnaire Model to a Process Model.

terms of BPMN 2.0 (Business Process Model and Notation) (Business Process Model, 2011). To establish the link between process and questionnaire model, we annotated the depicted graph with additional labels.

The processing of the questionnaire starts with the execution of activity *Page Intro*, which presents an introductory text for the participant interacting with the electronic questionnaire. This introduction includes, for example, instructions on how to fill in the questionnaire or how to interact with the smart mobile device. After completing this first step, activity *Page General* becomes enabled. In this form-based activity, data elements *Cigarettes*, *Drugs* and *Alcohol* are written. More precisely, the values of these data elements correspond to the answers given for the questions displayed on the respective page of the questionnaire. For example, the question corresponding to data element *Cigarettes* is as follows: “Do you smoke?” (with the possible answers “yes / no”). After completing activity *Page General*, an AND gateway (ANDsplit) becomes enabled. In turn, all outgoing paths of this ANDsplit (i.e., parallel split node) become enabled and are then executed concurrently. In the given application scenario, each of these paths contains an XOR Gateway (XORsplit), which reads one of the aforementioned data elements to make a choice among its outgoing paths. For example, assume that in *Page General* the participant has an

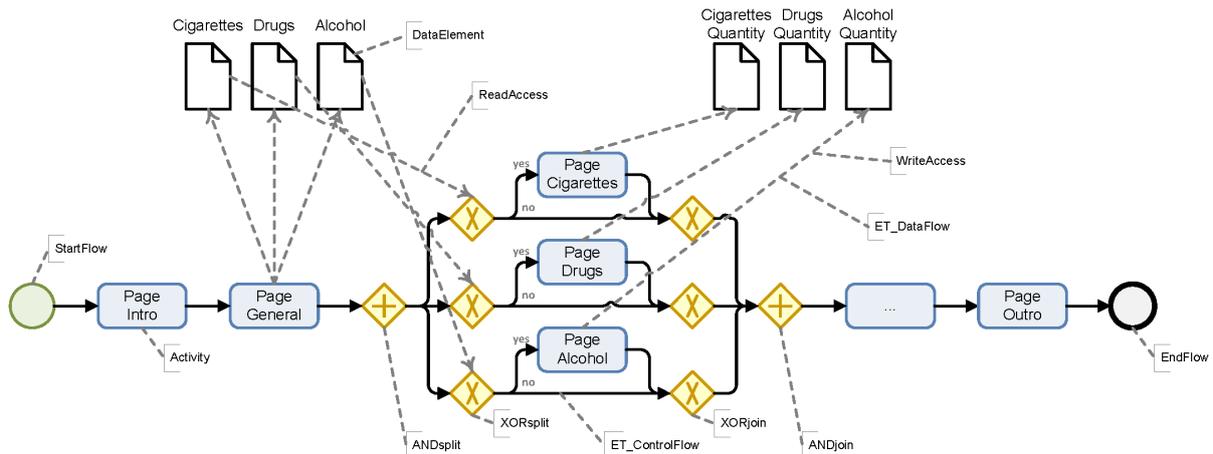


Figure 3: Application Scenario: an abbreviated Questionnaire with Annotations.

swered question “Do you smoke?” with “yes”. Then, in the respective XOR split, the upper path (labeled with “yes”) will be chosen, which consists of exactly one activity, i.e., *Page Cigarettes*. In the context of this activity, additional questions regarding the consumption of cigarettes will be displayed to the actor. This activity and page, respectively, is exemplarily displayed in Figure 4. Assume further that question “Do you take drugs? (yes / no)” has been answered with “no” in the context of *Page General*. Then, activity *Page Drugs* will be skipped as the lower path (labeled with “no”) of the respective XOR split will be chosen. As soon as all three branches are completed, the *ANDjoin* will become enabled and the succeeding activity be displayed. We omit further descriptions for activities of the questionnaire model due to lack of space. Finally, the processing of a questionnaire ends with activity *Page Outro*. Note that an arbitrary number of questionnaire instances processed by different participants may be created.

Figure 4 gives an impression of the *Page Cigarettes* activity. It displays additional questions regarding the consumption of cigarettes. This page is layouted automatically by the electronic questionnaire application based on the specified process model, which includes the pages to be displayed (cf. Figure 3). Note, that the data elements are used to create the user interface, as they contain the actual text of the questions as well as the possible answers to be displayed (i.e., the answers among which the user may choose).

3.3 Requirements for Process-based Questionnaires

When using process management technology to coordinate the collection of data with smart mobile de-

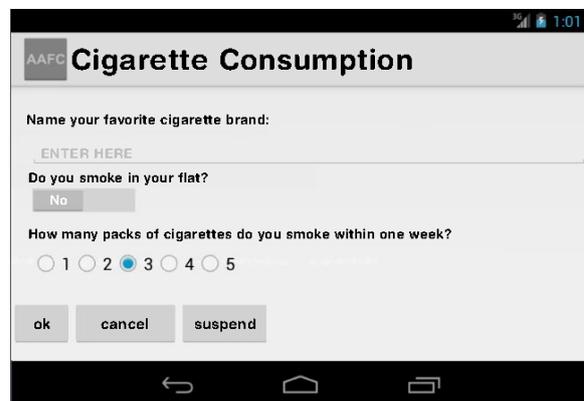


Figure 4: Activity “Page Cigarettes”.

vices, additional challenges emerge. In particular, these are related to the modeling of a questionnaire as well as the process-driven execution of corresponding questionnaire instances on smart mobile devices.

Since questionnaire-based interviews are often interactive, the participating roles (e.g., interviewer and interviewed subject) should be properly assisted when interacting with the smart mobile device. For example, it should be possible for them to start or abort questionnaire instances. In the context of long-running questionnaire instances, in addition, it might be required to interrupt an interview and continue it later. For this purpose, it must be possible to suspend the execution of a questionnaire instance and to resume it at a later point in time (with access to all data and answers collected so far). In the context of long-running interviews, it is also useful to be able to display an entire questionnaire and process model respectively. Therefore, already answered questions should be displayed differently (e.g., in a different color) compared to upcoming questions. Note that this is crucial for providing a quick overview about

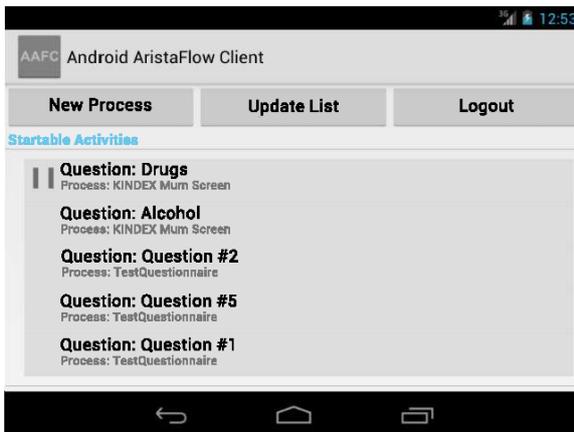


Figure 5: Startable Activities for a Specific Actor.

the progress of an interview.

Since domain experts might not be familiar with existing process modeling notations like BPMN 2.0, an easy-to-understand, self-explaining, and domain-specific process notation is needed. In addition, the roles participating in a questionnaire should be provided with specific views on the process model (i.e., questionnaire), e.g., hiding information not required for this role (Kolb and Reichert, 2013). For example, a subject may not be allowed to view subsequent questions in order to ensure credibility of the given answers.

Regarding the execution of the activities of a questionnaire (i.e., pages) additional challenges emerge.

The questions of a (psychological) questionnaire may have to be answered by different actors each of them possessing a specific role. For example, follow-up questions related to the subject involved in a psychological questionnaire might have to be answered by a psychologist and not by the subject itself. Consequently, the electronic questionnaire application must ensure that only those questions are displayed to an actor intended for him or her. Figure 5 shows the startable activities, currently available for a specific actor using the smart mobile device.

In many scenarios, the questions of an electronic questionnaire may have to be displayed together with possible answers. In order to avoid bad quality of the data collected, actors should be further assisted when interacting with the smart mobile device; e.g., through error messages, help texts, or on-the-fly validations of entered data.

To foster the subsequent analysis of the data collected, the latter needs to be archived in a central repository. Furthermore, additional information (e.g., the time it took the subject to answer a particular question) should be recorded in order to increase the expressiveness of the data collected. Finally,

anonymization of this data might have to be ensured as questionnaires often collect personal data and privacy constitutes a crucial issue. In certain cases, it might also become necessary to dismiss the results of an already answered question.

Taking these general requirements into account, we designed an architecture for an electronic questionnaire application.

4 ARCHITECTURE AND IMPLEMENTATION

This section introduces the architecture we developed for realizing mobile electronic questionnaires. In particular, the latter run on smart mobile devices and interact with a remote process engine. This architecture is presented in Section 4.1. Since this paper focuses on the requirements, challenges and lessons learned when applying state-of-the-art process management technology to realize electronic questionnaires, we will not describe the architecture of the process management system (and its process engine) in detail; see (Dadam and Reichert, 2009; Reichert et al., 2009) for respective work. The general architecture of our electronic questionnaire application is depicted in Figure 6.

4.1 Electronic Questionnaire Application

The electronic questionnaire application is divided into three main packages, which are related to the *user interface* ①, the *communication* ② with the external process engine, and useful *tools* for interacting with the client ③.

The user interface representing a particular page of the questionnaire is represented by an *ActivityTemplate* ④, which provides basic methods for the questionnaire (e.g., to start or stop an activity). In turn, the *LoginView* ⑤ is used to query the user credential and to select an available role for this actor (e.g., *name = JohnDoe, role = Interviewer*). Furthermore, the *MainView* ⑥ provides a list (e.g., worklist) with the pages currently available for the user interacting with the questionnaire. These list items are represented using the *ProcessAdapter* ⑦. Since the user interface of a questionnaire is generated dynamically depending on the underlying process model deployed on the process engine at runtime, a user interface generator is needed. This service is provided by the *ActivityView* ⑧. To interact with the device, different classes of the *Input* ⑨ elements used within a ques-

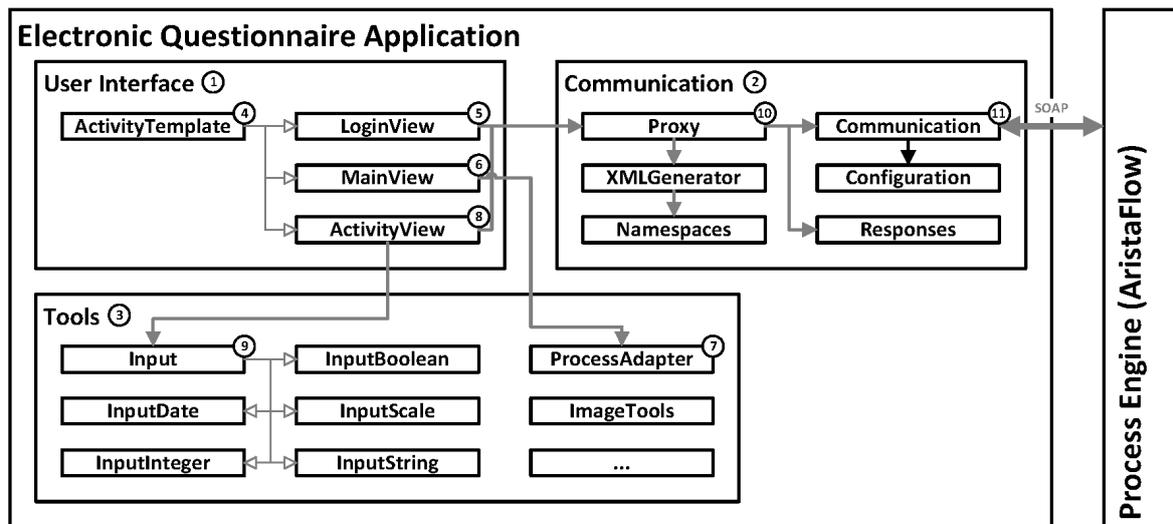


Figure 6: Architecture of the questionnaire application.

tionnaire are provided. These classes provide the necessary logic to interact with the input elements as well as the corresponding graphical representation of this element. As certain input elements are platform-specific (e.g., there is no spinning wheel for standard desktop applications), missing input elements might be rendered differently, depending on the underlying platform (e.g., the spinning wheel on the iOS platform could be rendered as a dropdown element on a normal computer).

The complete communication with the external process engine should be handled by a *Proxy* ⑩ service. The latter is capable of generating the necessary request messages, which are then converted to SOAP request messages by the *Communication* ⑪ service and sent to the process engine. The response messages (e.g., the next page to display) sent by the process engine are then received by the *Communication* and decomposed by the *Proxy*. Afterwards, the data within this message is visualized in the *ActivityView*, which includes the already mentioned user interface generator as well.

4.2 Proof-of-Concept Prototype

To validate the feasibility of the described architecture as well as to be able to apply it in a real setting, we implemented a proof-of-concept prototype for the Android platform. We decided to use the latter, since it is easier to generate and handle SOAP (Curbera et al., 2002) calls within the application compared to iOS. The prototype application was then used to verify the prescribed mental model (cf. Section 3), and to detail the requirements regarding the execution of process-aware questionnaires. Furthermore,



Figure 7: Different question types and their visualization within the questionnaire client application.

additional insights into the practical use of this electronic questionnaire application by domain experts in the context of their studies could be gathered. We were able to meet the requirements presented in Section 2.2 when implementing the questionnaire client application, even though certain drawbacks still exist. To enable domain experts, who usually have no programming skills, to create a mobile electronic questionnaire, we implemented a fully automated user interface generator for the mobile application itself. In addition, we were able to provide common types for questions used within a questionnaire (e.g., likert-scale, free-text, or yes-no-switches). These types are automatically mapped to appropriate input elements visualized within the application. Figure 7 gives an impression of the input elements implemented.

5 DISCUSSION

This section discusses our approach and reflects on the experiences we gained when applying state-of-the-art process management technology to support mobile data collection with electronic questionnaires. Since we applied an implemented questionnaire in a psychological study, we were also able to gain addi-

tional insights into practical issues.

The presented approach has focused on the development of smart mobile device applications enabling flexible data collection rather than on the design of a development framework. Therefore, we have used an existing process modeling editor for defining the process logic of electronic questionnaires. However, since the domain experts using our questionnaire application have been unfamiliar with the BPMN 2.0 modeling notation, a number of training sessions were required to make them familiar with BPMN 2.0. Afterwards, they were able to create their own questionnaires. In particular, the abstraction introduced by the use of process models for specifying the logic of questionnaires was well understood by domain experts. However, the training sessions have shown that there is a need for a more user-friendly, domain-specific modeling notation, enabling domain experts to define questionnaires on their own. In particular, such a domain-specific modeling language needs to be self-explaining and easy to use. Further, it should hide modeling elements not required in the given use case. Note that BPMN 2.0 provides many elements not needed for defining the logic of electronic questionnaires. Consider, for example, the AND-gateways, which allow modeling the parallel execution. Regarding the use case of mobile data collection, it does not matter which path is going to be evaluated first. On the other hand, elements used for modeling the questionnaire should have a meaningful and expressive representation. Thus, an activity should be represented as page-symbol to add more context-aware information to the questionnaire model.

As we further learned in our case study, the creation and maintenance of a questionnaire constitutes a highly interactive, flexible and iterative task. In general, the editing of already existing, but not yet published questionnaires, should be self-explaining. Basic patterns dealing with the adaptation of the logic of a questionnaire (e.g., moving a question to another position or adding a new question) should be integrated in a modeling editor to provide tool-support for creating and managing questionnaires.

As discussed in Section 3.3, we use process management technology for modeling and enacting electronic questionnaires. Accordingly the created questionnaire model needs to be deployed on a process engine. Regarding the described client server architecture (cf. Section 4.1), all process (i.e., questionnaire) models are stored and executed on the server running the process engine. Keeping in mind that mobile questionnaires might be also used in areas without stable internet connection, any approach requiring a permanent internet connection between the mobile

client and the process engine running on an external server will not be accepted. In order to cope with this issue, a light-weight process engine is required, which can run on the smart mobile device. We have started working in this direction as well; e.g., see (Pryss et al., 2010a; Pryss et al., 2010b).

Since the user interface of the electronic questionnaire is automatically generated based on the provided process model, the possibilities to customize the layout of a questionnaire are rather limited. From the feedback we had obtained from domain experts, however, it became clear that an expressive layout component is needed that allows controlling the visual appearance of a questionnaire running on smart mobile devices. Among others, different text styles (e.g., bold), spacing between input elements (e.g., bottom spacing), and absolute positioning of elements should be supported. In addition, the need for integrating images has been expressed several times.

Since we use process-driven electronic questionnaires for collecting data with smart mobile devices, the answers provided by the actors filling in the questionnaire could be directly transferred to the server. This will relieve the actors from time-consuming manual tasks. Furthermore, as there exists a process model describing the flow logic of the questionnaire as well as comprehensive instance data (e.g., instance execution history), process mining techniques for analyzing questionnaire instances may be applied (van der Aalst et al., 2007). In addition, Business Intelligence Systems (Anandarajan et al., 2003) could reveal further interesting aspects with respect to the data collected in order to increase the expressiveness of the analysis. Such systems would allow for a faster evaluation and relieve domain experts from manual tasks such as transferring the collected data into electronic worksheets.

Finally, we have experienced a strong acceptance among all participating actors (e.g., interviewers, domain experts, and subjects) regarding the practical benefits of electronic questionnaire applications on smart mobile devices. Amongst others this was reflected by a much higher willingness to fill out an electronic questionnaire compared to the respective paper-based version (Ruf-Leuschner et al., 2013; Isele et al., 2013). Furthermore, a higher motivation to complete the questionnaire truthfully could be observed. Of course, this acceptance partly results from the modern and intuitive way to interact with smart mobile devices.

6 RELATED WORK

There exists a variety of questionnaire systems available on the market. In general, these systems can be classified into two groups: *online services* (SurveyMonkey, 2013) and *standalone applications* (Electric Paper Evaluationssysteme, 2013). Due to the fact that a questionnaire might contain sensitive information (e.g., the mental status of a subject or personal details), online surveys are often not appropriate for this type of data collection applications. As another limitation of online systems, local authorities do often not allow third-party software systems to store the information of a subject. However, these applications also must deal with privacy issues. Standalone applications usually offer possibilities to create a questionnaire, but do not deal with the requirements discussed in this paper. Furthermore, they lack a flexible and mobile data collection. Usually, respective questionnaires are displayed as web applications, which cannot be used when no internet connection is available.

To the best of our knowledge, the process-aware enactment of questionnaires on smart mobile devices has not been considered comprehensively by other groups so far. In previous studies, we identified crucial issues regarding the implementation of psychological questionnaires on smart mobile devices (Liebrecht, 2012; Schindler, 2013). In these studies, we aimed at preserving the validity of psychological instruments, which is a crucial point when replacing paper-based questionnaires with electronic ones. Although the implemented applications have already shown several advantages in respect to data collection and analysis, they have not been fully suitable for realizing psychological questionnaires in the large scale yet. In particular, maintenance efforts for domain experts and other actors were considerably high. More precisely, changes of an implemented questionnaire (or its structure) still had to be accomplished by computer scientists, since its implementation is hard-coded. Therefore aim at the integration of a process-aware information system to overcome this limitation.

Focusing on the complete lifecycle of paper-based questionnaires and supporting every phase with mobile technology has actually not been considered by other groups so far. However, there exists some work related to mobile data collection. In particular, mobile process management systems, as described in (Pryss et al., 2013; Wakholi et al., 2011; Kunze et al., 2007), could be used to realize electronic questionnaires. However, this use case has not been considered by respective mobile process engines so far.

The QuON platform (Paul et al., 2013) provides a web-based survey system, which provides a variety

of different input types for the questions used within a questionnaire. QuON does not use a model-based representation to specify a questionnaire as in our approach. Another distinctive characteristic of QuON is the webbased-only approach. Especially in psychological field studies, the latter will result in problems as the QuON platform does not use responsive web-design.

Movilitas (Movilitas, 2013) applies SAP Sybase Unwired Platform to enable mobile data collection for business scenarios. The Sybase Unwired Platform is a highly adaptive implementation framework for mobile applications, which directly interacts with a backend, providing all required business data. Further research is required to show, whether this approach can be used to realize mobile electronic questionnaires in domains like psychology or health care as well.

Finally, (Kolb et al., 2012) present a set of patterns for expressing user interface logic based on the same notation as used for business process modeling. In particular, a transformation method is introduced, which applies these patterns to automatically derive user interfaces by establishing a bidirectional mapping between process model and user interface.

7 SUMMARY & OUTLOOK

In this paper, limitations of paper-based questionnaires for data collection were discussed. To deal with these limitations, we derived characteristic requirements for electronic questionnaire applications. In order to meet these requirements, we suggested the use of process management technology. According to the mental model introduced, a questionnaire and its logic can be described in terms of a process model at a higher level of abstraction. To evaluate our approach, a sophisticated application scenario from the psychological domain was considered. We have shown how a questionnaire can be mapped to a process model.

In the interviews we conducted with domain experts as well as from other implemented business applications we elaborated general requirements for flexible mobile data collection on smart mobile devices. These cover major aspects such as the secure and encrypted communication. Note that the latter is crucial, especially in the medical and psychological domains, which both deal with sensitive information of the subjects involved. We further presented an architecture enabling such mobile data collection applications based on a smart mobile device and a process engine. As another contribution, we demonstrated the feasibility of our proof-of-concept application. Several features as well as problems regard-

ing the implementation and communication with the server component, hosting the process engine, have been highlighted. Finally, we discussed the benefits of using process-aware questionnaire application for mobile data collection.

In future work, we plan to extend our approach with additional features. First, we will provide a mobile process engine running on the smart mobile device itself. This will enable a process-driven enactment of questionnaire instances even if no permanent internet connection is available. We consider this as a fundamental feature for enabling flexible data collection applications on smart mobile devices. However, this will be accompanied with other problems, such as the proper synchronization among multiple devices (e.g., if changes were made to the model of the questionnaire) in order to keep the devices at the same level of information. In addition, we want to conceptualize a *generic questionnaire system*, which is able to support the complete lifecycle of a questionnaire. To disseminate this system among domain experts being unfamiliar and unaware of modeling process logic with standard notations, in addition, an easy to understand, but still precise notation for defining process-aware questionnaires is needed. To further enhance data analysis capabilities (e.g., further analysis of the given answers), we have started to integrate sensors measuring vital signs in order to gather other information about subjects during interviews (Schobel et al., 2013). As a major benefit of the framework, we expect higher data quality, shorter evaluation cycles and a significant decrease in workload. In particular, it enables a high level of abstraction in defining electronic questionnaires that may run on smart mobile devices.

ACKNOWLEDGEMENT

Supported by funds from the program *Research initiatives, infrastructure, network and transfer platforms* in the framework of the *DFG Excellence Initiative - Third Funding Line*.

REFERENCES

- Anandarajan, M., Anandarajan, A., and Srinivasan, C. A. (2003). *Business intelligence techniques: a perspective from accounting and finance*. Springer.
- Business Process Model (2011). Business Process Model and Notation (BPMN) Version 2.0. *OMG Specification, Object Management Group*.
- Crombach, A., Nandi, C., Bambonye, M., Liebrecht, M., Pryss, R., Reichert, M., Elbert, T., and Weierstall, R. (2013). Screening for mental disorders in post-conflict regions using computer apps - a feasibility study from burundi. In *XIII Congress of European Society of Traumatic Stress Studies (ESTSS) Conference*.
- Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N., and Weerawarana, S. (2002). Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI. *IEEE, Internet Computing*, 6(2):86–93.
- Dadam, P. and Reichert, M. (2009). The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support - Challenges and Achievements. *Computer Science - Research and Development*, 23(2):81–97.
- Electric Paper Evaluationssysteme (2013). EvaSys. <http://www.evsys.de/>. last visited: 05. November 2013.
- Isele, D., Ruf-Leuschner, M., Pryss, R., Schauer, M., Reichert, M., Schobel, J., Schindler, A., and Elbert, T. (2013). Detecting adverse childhood experiences with a little help from tablet computers. In *XIII Congress of European Society of Traumatic Stress Studies (ESTSS) Conference*.
- Kolb, J., Hübner, P., and Reichert, M. (2012). Automatically Generating and Updating User Interface Components in Process-Aware Information Systems. In *20th Int'l Conference on Cooperative Information Systems*, number 7565 in LNCS, pages 444–454. Springer.
- Kolb, J. and Reichert, M. (2013). A flexible approach for abstracting and personalizing large business process models. *Applied Computing Review*, 13(1):6–17.
- Kunze, C. P., Zaplata, S., and Lamersdorf, W. (2007). Mobile processes: Enhancing cooperation in distributed mobile environments. *Journal of Computers*, 2(1):1–11.
- Liebrecht, M. (2012). Technische Konzeption und Realisierung einer mobilen Anwendung für den Konstanzer Index zur Erhebung von psychosozialen Belastungen während der Schwangerschaft. Diploma Thesis, University of Ulm.
- Movilitas (2013). Movilitas Consulting AG. <http://www.movilitas.com/>. last visited: 04. November 2013.
- Paul, D., Wallis, M., Henskens, F., and Nolan, K. (2013). QuON: A Generic Platform for the Collation and Sharing of Web Survey Data. *International Conference on Web Information Systems and Technologies*.
- Pryss, R., Langer, D., Reichert, M., and Hallerbach, A. (2012). Mobile Task Management for Medical Ward Rounds - The MEDo Approach. In *1st Int'l Workshop on Adaptive Case Management (ACM'12), BPM'12 Workshops*, number 132 in LNBIP, pages 43–54. Springer.
- Pryss, R., Musiol, S., and Reichert, M. (2013). Collaboration Support Through Mobile Processes and Entailment Constraints. In *9th IEEE Int'l Conference on Collaborative Computing: Networking, Applications and Worksharing*. IEEE Computer Society Press.
- Pryss, R., Tiedeken, J., Kreher, U., and Reichert, M. (2010a). Towards flexible process support on mobile

- devices. In *Proc CAiSE'10 Forum - Information Systems Evolution*, number 72 in LNBIP, pages 150–165. Springer.
- Pryss, R., Tiedeken, J., and Reichert, M. (2010b). Managing Processes on Mobile Devices: The MARPLE Approach. In *CAiSE'10 Demos*.
- Reichert, M. and Dadam, P. (1998). ADEPTflex-Supporting Dynamic Changes of Workflows Without Losing Control. *Journal of Intelligent Information Systems, Special Issue on Workflow Management Systems*, 10(2):93–129.
- Reichert, M. and Dadam, P. (2009). Enabling Adaptive Process-aware Information Systems with ADEPT2. In Cardoso, J. and van der Aalst, W., editors, *Handbook of Research on Business Process Modeling*, pages 173–203. Information Science Reference, Hershey, New York.
- Reichert, M., Dadam, P., Rinderle-Ma, S., Jurisch, M., Kreher, U., and Goeser, K. (2009). Architectural Principles and Components of Adaptive Process Management Technology. In *PRIMIUM - Process Innovation for Enterprise Software*, number P-151 in Lecture Notes in Informatics (LNI), pages 81–97.
- Reichert, M. and Weber, B. (2012). *Enabling Flexibility in Process-Aware Information Systems: Challenges, Methods, Technologies*. Springer, Berlin-Heidelberg.
- Robecke, A., Pryss, R., and Reichert, M. (2011). DBIScholar: An iPhone Application for Performing Citation Analyses. In *CAiSE Forum-2011*, volume Vol-73 of *Proc CAiSE'11 Forum*. CEUR Workshop Proceedings.
- Ruf-Leuschner, M., Pryss, R., Liebrecht, M., Schobel, J., Spyridou, A., Reichert, M., and Schauer, M. (2013). Preventing further trauma: KINDEX mum screen - assessing and reacting towards psychosocial risk factors in pregnant women with the help of smartphone technologies. In *XIII Congress of European Society of Traumatic Stress Studies (ESTSS) Conference*.
- Schindler, A. (2013). Technische Konzeption und Realisierung des MACE-Tests mittels mobiler Technologie. Bachelor Thesis, University of Ulm.
- Schobel, J., Schickler, M., Pryss, R., Nienhaus, H., and Reichert, M. (2013). Using Vital Sensors in Mobile Healthcare Business Applications: Challenges, Examples, Lessons Learned. In *9th Int'l Conference on Web Information Systems and Technologies (WEBIST 2013), Special Session on Business Apps*, pages 509–518.
- SurveyMonkey (2013). SurveyMonkey: Free online survey software & questionnaire tool. <http://www.surveymonkey.com/>. last visited: 14. May 2013.
- van der Aalst, W. M., Reijers, H. A., Weijters, A. J., van Dongen, B. F., Alves de Medeiros, A., Song, M., and Verbeek, H. (2007). Business process mining: An industrial application. *Information Systems*, 32(5):713–732.
- Wakholi, P., Chen, W., and Klungsøyr, J. (2011). Workflow support for mobile data collection. In *Enterprise, Business-Process and Information Systems Modeling*, pages 299–313. Springer.
- Weber, B., Reichert, M., Mendling, J., and Reijers, H. (2011). Refactoring large process model repositories. *Computers in Industry*, 62(5):467–486.

Location-based Mobile Augmented Reality Applications

Challenges, Examples, Lessons Learned

Philip Geiger, Marc Schickler, Rüdiger Pryss, Johannes Schobel and Manfred Reichert
Institute of Databases and Information Systems, University of Ulm, James-Franck-Ring, Ulm, Germany
{philip.geiger, marc.schickler, ruediger.pryss, johannes.schobel, manfred.reichert}@uni-ulm.de

Keywords: Smart Mobile Applications, Location-based Mobile Augmented Reality.

Abstract: The technical capabilities of modern smart mobile devices more and more enable us to run desktop-like applications with demanding resource requirements in mobile environments. Along this trend, numerous concepts, techniques, and prototypes have been introduced, focusing on basic implementation issues of mobile applications. However, only little work exists that deals with the design and implementation (i.e., the engineering) of advanced smart mobile applications and reports on the lessons learned in this context. In this paper, we give profound insights into the design and implementation of such an advanced mobile application, which enables location-based mobile augmented reality on two different mobile operating systems (i.e., iOS and Android). In particular, this kind of mobile application is characterized by high resource demands since various sensors must be queried at run time and numerous virtual objects may have to be drawn in realtime on the screen of the smart mobile device (i.e., a high frame count per second be caused). We focus on the efficient implementation of a robust mobile augmented reality engine, which provides location-based functionality, as well as the implementation of mobile business applications based on this engine. In the latter context, we also discuss the lessons learned when implementing mobile business applications with our mobile augmented reality engine.

1 INTRODUCTION

Daily business routines increasingly require access to information systems in a mobile manner, while requiring a desktop-like feeling of mobile applications at the same time. However, the design and implementation of mobile business applications constitutes a challenging task (Robecke et al., 2011). On one hand, developers must cope with limited physical resources of smart mobile devices (e.g., limited battery capacity or limited screen size) as well as non-predictable user behaviour (e.g., mindless instant shutdowns). On the other, mobile devices provide advanced technical capabilities, including motion sensors, a GPS sensor, and a powerful camera system. Hence, new types of business applications can be designed and implemented in the large scale. Integrating sensors and utilizing the data recorded by them, however, is a non-trivial task when considering requirements like robustness and scalability as well. Moreover, mobile business applications have to be developed for different mobile operating systems (e.g., iOS and Android) in order to allow for their widespread use. Hence, developers of mobile business applications must also cope with the heterogeneity of existing mobile operating systems, while at the same time utilizing their

technical capabilities. In particular, if mobile application users shall be provided with the same functionality in the context of different mobile operating systems, new challenges may emerge when considering scalability and robustness. This paper deals with the development of a generic mobile application, which enables location-based mobile augmented reality for realizing advanced business applications. We discuss the core challenges emerging in this context and report on the lessons learned when applying it to implement real-world mobile business applications. Existing related work has been dealing with location-based mobile augmented reality as well (Fröhlich et al., 2006; Carmigniani et al., 2011; Paucher and Turk, 2010; Reitmayr and Schmalstieg, 2003). To the best of our knowledge, they do not focus on aspects regarding the efficient integration of location-based mobile augmented reality with real-world mobile business applications.

1.1 Problem Statement

The overall purpose of this work is to show how to develop the core of a *location-based mobile augmented reality engine* for the mobile operating systems *iOS 5.1 (or higher)* and *Android 4.0 (or higher)*. We de-

note this engine as *AREA*¹. As a particular challenge, the augmented reality engine shall be able to display *points of interest (POIs)* from the surrounding of a user on the screen of his smart mobile device. In particular, POIs shall be drawn based on the *angle of view* and the *position* of the smart mobile device. This means that the real image captured by the camera of the smart mobile device will be augmented by *virtual objects (i.e., the POIs)* relative to the current position and attitude. The overall goal is to draw POIs on the camera view of the smart mobile device.

The development of a mobile augmented reality engine constitutes a non-trivial task. In particular, the following challenges emerge:

- In order to enrich the image captured by the smart mobile device's camera with virtual information about POIs in the surrounding, basic concepts enabling location-based calculations need to be developed.
- An efficient and reliable technique for calculating the distance between two positions is required (e.g., based on data of the GPS sensor in the context of outdoor location-based scenarios).
- Various sensors of the smart mobile device must be queried correctly in order to determine the attitude and position of the smart mobile device.
- The angle of view of the smart mobile device's camera lens must be calculated to display the virtual objects on the respective position of the camera view.

Furthermore, a location-based mobile augmented reality engine should be provided for all established mobile operating systems. However, to realize the same robustness and ease-of-use for heterogeneous mobile operating systems, is a non-trivial task.

1.2 Contribution

In the context of *AREA*, we developed various concepts for coping with the limited resources on a smart mobile device, while realizing advanced features with respect to mobile augmented reality at the same time. In this paper, we present a sophisticated application architecture, which allows integrating augmented reality with a wide range of applications. However, this architecture must not neglect the characteristics of the underlying kind of mobile operating system. While in many scenarios the differences between mobile operating systems are rather uncrucial when implementing

¹*AREA* stands for Augmented Reality Engine Application. A video demonstrating *AREA* can be viewed at: <http://vimeo.com/channels/434999/63655894>. Further information can be found at: <http://www.area-project.info>

a mobile business application, for the present mobile application this does no longer apply. Note that there already exist augmented reality frameworks and applications for mobile operating systems like Android or iOS. These include proprietary and commercial engines as well as open source frameworks and applications (Lee et al., 2009; Wikitude, 2013). To the best of our knowledge, however, these proposals neither provide insights into the functionality of such an engine nor its customization to a specific purpose. Furthermore, insights regarding the development of engines running on more than one mobile operating systems are usually not provided. To remedy this situation, we report on the lessons learned when developing *AREA* and integrating it with our mobile business applications.

This paper is organized as follows: Section 2 introduces core concepts and the architecture of *AREA*. In Section 3, we discuss lessons learned when implementing *AREA* on the iOS and Android mobile operating systems. In particular, this section discusses differences we experienced in this context. Section 4 gives detailed insights into the use of *AREA* for implementing real-world business applications. In Section 5 related work is discussed. Section 6 concludes the paper with a summary and outlook.

2 AREA APPROACH

The basic concept realized in *AREA* is the *locationView*. The points of interest inside the camera's field of view are displayed on it, having a size of $\sqrt{width^2 + height^2}$ pixels. The *locationView* is placed centrally on the screen of the mobile device.

2.1 The locationView

Choosing the particular approach provided by the *locationView* has specific reasons, which we discuss in the following.

First, *AREA* shall display *points of interest (POIs)* correctly, even if the device is held obliquely. Depending on the device's attitude, the POIs then have to be rotated with a certain angle and moved relatively to the rotation. Instead of rotating and moving every POI separately in this context, however, it is also possible to only rotate the *locationView* to the desired angle, whereas the POIs it contains are rotated automatically; i.e., resources needed for complex calculations can be significantly reduced.

Second, a complex recalculation of the field of view of the camera is not required if the device is

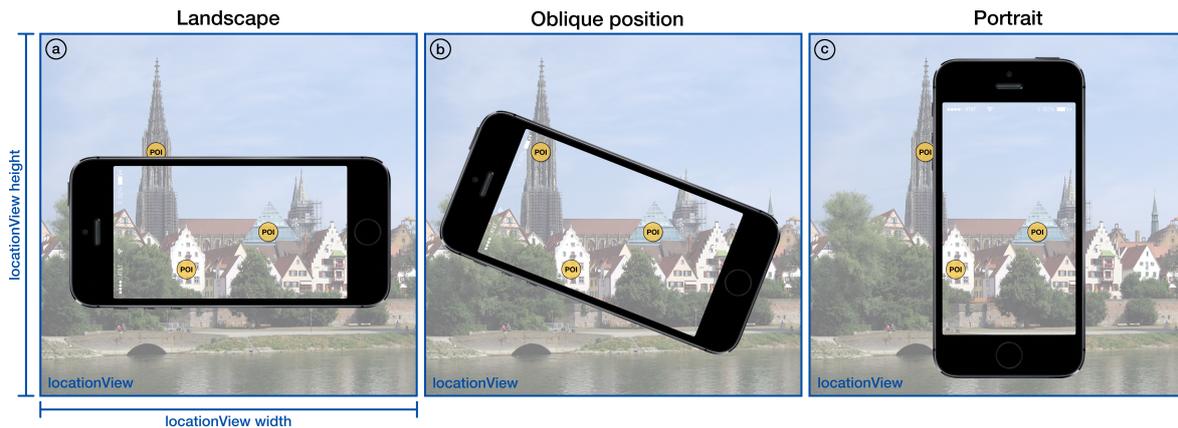


Figure 1: Examples of locationView depicting its characteristics.

in an oblique position. The vertical and horizontal dimensions of the field of view are scaled proportionally to the diagonal of the screen, such that a new maximum field of view results with the size of $\sqrt{width^2 + height^2}$ pixels. Since the *locationView* is placed centrally on the screen, the camera's actual field of view is not distorted. Further, it can be customized by rotating it contrary to the rotation of the device. The calculated maximal field of view is needed to efficiently draw POIs visible in portrait mode, landscape mode, or any oblique position inbetween.

Fig. 1 presents an example illustrating the concept of the *locationView*. Thereby, each sub-figure represents one *locationView*. As one can see, a *locationView* is bigger than the display of the respective mobile device. Therefore, the camera's field of view must be increased by a certain factor such that all POIs, which are either visible in portrait mode (cf. Fig. 1c), landscape mode (cf. Fig. 1a), or any rotation inbetween (cf. Fig. 1b), are drawn on the *locationView*. For example, Fig. 1a shows a POI (on the top) drawn on the *locationView*, but not yet visible on the screen of the device in landscape mode. Note that this POI is not visible for the user until he rotates his device to the position depicted in Fig. 1b. Furthermore, when rotating the device from the position depicted in Fig. 1b to portrait mode (cf. Fig. 1c), the POI on the left disappears again from the field of view, but still remains on the *locationView*.

The third reason for using the presented *locationView* concept concerns performance. When the display has to be redrawn, the POIs already drawn on the *locationView* can be easily queried and reused. Instead of first clearing the entire screen and afterwards re-initializing and redrawing already visible POIs, POIs that shall remain visible, do not have to be redrawn. Furthermore, POIs located outside the field

of view after a rotation are deleted from it, whereas POIs that emerge inside the field of view are initialized.

Fig. 2 sketches the basic algorithm used for realizing this *locationView*².

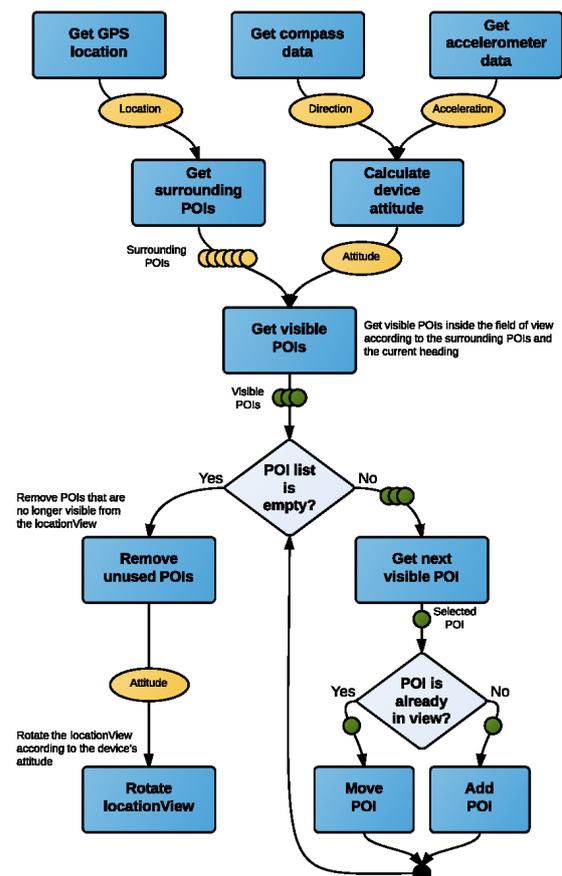


Figure 2: Algorithm realizing the locationView (sketched).

²More technical details can be found in a technical report (Geiger et al., 2013)

2.2 Architecture

The AREA architecture has been designed with the goal to be able to easily exchange and extend its components. The design comprises four main modules organized in a multi-tier architecture and complying with the *Model View Controller* pattern (cf. Fig. 3). Lower tiers offer their services and functions by interfaces to upper tiers. In particular, the red tier (cf. Fig. 3) will be described in detail in Section 3, when discussing the differences regarding the development of AREA on the iOS and Android platforms. Based on this architectural design, modularity can be ensured; i.e., both data management and various elements (e.g., the POIs) can be customized and extended on demand. Furthermore, the compact design of AREA enables us to build new mobile business applications based on it as well as to easily integrate it with existing applications.

The lowest tier, called *Model*, provides modules and functions to exchange the POIs. In this context, we use both an *XML*- and a *JSON*-based interface to collect and parse POIs. In turn, these POIs are then stored in a global database. Note that we do not rely on the *ARML* schema (ARML, 2013), but use our own *XML* schema. In particular, we will be able to extend our *XML*-based format in the context of future research on AREA. Finally, the *JSON* interface uses a light-weight, easy to understand, and extendable format with which developers are familiar.

The next tier, called *Controller*, consists of two main modules. The *Sensor Controller* is responsible for culling the sensors necessary to determine the device's location and orientation. The sensors to be culled include the *GPS sensor*, the *accelerometer*, and the *compass sensor*. The *GPS sensor* is used to determine the position of the device. Since we currently focus on location-based outdoor scenarios, *GPS* coordinates are predominantly used. In future work, we address indoor scenarios as well. Note that the architecture of AREA has been designed to easily change the way coordinates will be obtained. Using the *GPS* coordinates and its corresponding altitude, we can calculate the distance between mobile device and *POI*, the horizontal bearing, and the vertical bearing. The latter is used to display a *POI* higher or lower on the screen, depending on its own altitude. In turn, the *accelerometer* provides data for determining the current rotation of the device, i.e., the orientation of the device (landscape, portrait, or any other orientation inbetween) (cf. Fig. 1). Since the *accelerometer* is used to determine the vertical viewing direction, we need the *compass* data of the mobile device to determine the horizontal viewing direction of the user as

well. Based on the vertical and horizontal viewing directions, we are able to calculate the direction of the field of view as well as its boundaries according to the camera angle of view of the device. The *Point of Interest Controller* (cf. Fig. 3) uses data of the *Sensor Controller* in order to determine whether a *POI* is inside the vertical and horizontal field of view. Furthermore, for each *POI* it calculates its position on the screen taking the current field of view and the camera angle of view into account.

The uppermost tier, called *View*, consists of various user interface elements, e.g., the *locationView*, the *Camera View*, and the specific view of a *POI* (i.e., the *Point of Interest View*). Thereby, the *Camera View* displays the data captured by the device's camera. Right on top of the *Camera View*, the *locationView* is placed. It displays *POIs* located inside the current field of view at their specific positions as calculated by the *Point of Interest Controller*. To rotate the *locationView*, the interface of the *Sensor Controller* is used. The latter allows to determining the orientation of the device. Furthermore, a radar can be used to indicate the direction in which invisible *POIs* are located (cf. Fig. 9 shows an example of the radar). Finally, AREA make use of libraries of the mobile development frameworks themselves, which provide access to core functionality of the underlying operating system, e.g., sensor access and screen drawing functions (cf. *Native Frameworks* in Fig. 3).

3 EXPERIENCES WITH IMPLEMENTING AREA ON EXISTING MOBILE OPERATING SYSTEMS

The kind of business application we consider utilizes the various sensors of smart mobile devices, and hence provides new kinds of features compared to traditional business applications. However, this significantly increases complexity for application developers as well. This complexity further increases if the mobile application shall be provided for different mobile operating systems.

Picking up the scenario of mobile augmented reality, this section gives insights into ways for efficiently handling the *POIs*, relevant for the *locationView* of our mobile augmented reality engine. In this context, the implementation of the *Sensor Controller* and the *Point of Interest Controller* are most interesting regarding the subtle differences one must consider when developing such an engine on different mobile operating systems (i.e., iOS and Android).

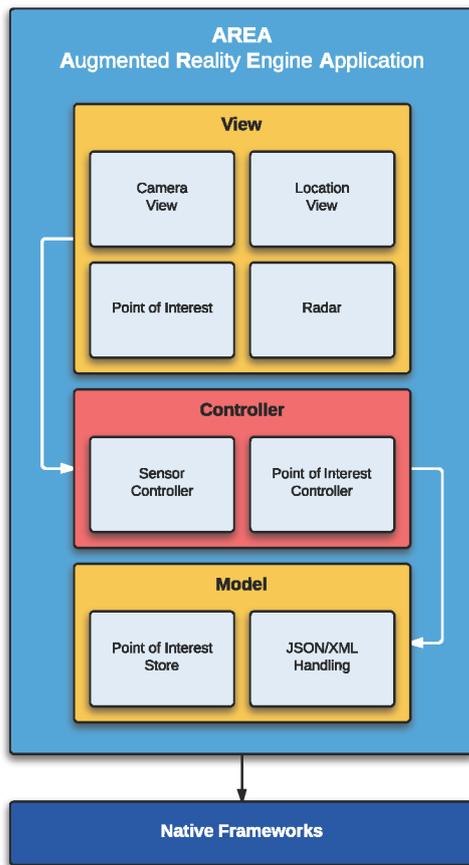


Figure 3: Multi-tier architecture of AREA.

In order to reach a high efficiency when displaying or redrawing POIs on the screen, we choose a native implementation of AREA on the iOS and Android mobile operating systems. Thus, we can make use of provided built-in APIs of these operating systems, and can call native functions without any translation as required in frameworks like *Phonegap* (Systems, 2013). Note that efficiency is very crucial for mobile business applications (Corral et al., 2012) since smart mobile devices rely on battery power. Therefore, to avoid high battery usage by expensive framework translations, only a native implementation is appropriate in our context. Apart from this, most cross-platform development frameworks do not provide a proper set of functions to work with sensors (Schobel et al., 2013). In the following, we present the implementation of AREA on both the iOS and the Android mobile operating systems.

3.1 iOS Mobile Operating System

The iOS version of AREA has been implemented using the programming language Objective-C and iOS

Version 7.0 on Apple iPhone 4S. Furthermore, for developing AREA, the Xcode environment (Version 5) has been used.

3.1.1 Sensor Controller

The *Sensor Controller* is responsible for culling the necessary sensors in order to correctly position the POIs on the screen of the smart mobile device. To achieve this, iOS provides the *CoreMotion* and *CoreLocation* frameworks. We use the *CoreLocation* framework to get notified about changes of the location as well as compass heading. Since we want to be informed about every change of the compass heading, we adjusted the heading filter of the *CoreLocation* framework accordingly. When the framework sends us new heading data, its data structure contains a real heading as well as a magnetic one as floats. The real heading complies to the geographic north pole, whereas the magnetic heading refers to the magnetic north pole. Since our coordinates corresponds to GPS coordinates, we use the real heading data structure. Note that the values of the heading will become (very) inaccurate and oscillate when the device is moved. To cope with this, we apply a *lowpass filter* (Kamenetsky, 2013) to the heading in order to obtain smooth and accurate values, which can then be used to position the POIs on the screen. Similar to the heading, we can adjust how often we want to be informed about location changes. On one hand, we want to get notified about all relevant location changes; on the other, every change requires a recalculation of the surrounding POIs. Thus, we decided to get notified only if a difference of at least 10 meters occurs between the old and the new location. Note that this is generally acceptable for the kind of applications we consider (cf. Section 4.1). Finally, the data structure representing a location contains GPS coordinates of the device in degrees north and degrees east as decimal values, the altitude in meters, and a time stamp.

In turn, the *CoreMotion* framework provides interfaces to cull the accelerometer. The accelerometer is used to calculate the current rotation of the device as well as to determine in which direction the smart mobile device is pointing (e.g., in upwards or downwards direction). As opposed to location and heading data, accelerometer data is not automatically pushed by the iOS *CoreMotion* framework to the application. Therefore, we had to define an application loop that is polling this data every $\frac{1}{90}$ seconds. On one hand, this rate is fast enough to obtain smooth values; on the other, it is low enough to save battery power. As illustrated by Fig. 4, the data the accelerometer delivers consists of three values, i.e., the accelerations in x-, y-, and z-direction ((Apple, 2013)). Since grav-

ity is required for calculating in which direction a device is pointing, but we cannot obtain this gravity directly using the acceleration data, we had to additionally apply a *lowpass* filter (Kamenetsky, 2013), i.e., the filter is used for being applied to the x-, y-, and z-direction values. Thereby, the three values obtained are averaged and filtered. In order to obtain the vertical heading as well as the rotation of the device, we then have to apply the following steps: First, by calculating $\arcsin(z)$, we obtain a value between $\pm 90^\circ$ and describing the vertical heading. Second, by calculating $\arctan 2(-y, x)$, we obtain a value between 0° and 359° , describing the degree of the amount of the rotation of the (Alasdair, 2011) of the device.

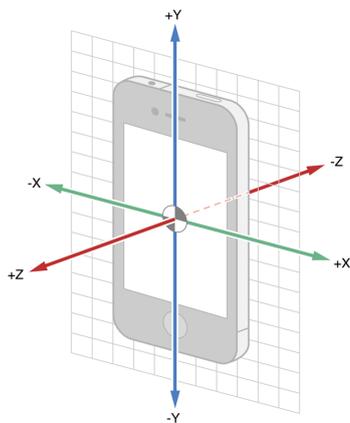


Figure 4: The three axes of the iPhone acceleration sensor (Apple, 2013).

Since we need to consider all possible orientations of the smart mobile device, we must adjust the compass data accordingly. For example, assume that we hold the device in portrait mode in front of us towards the North. Then, the compass data we obtain indicate that we are viewing in northern direction. As soon as we rotate the device, however, the compass data will change, although our view still goes to northern direction. This is caused by the fact that the reference point of the compass corresponds to the upper end of the device. To cope with this issue, we must adjust the compass data using the above presented rotation calculation. When subtracting the rotation value (i.e., 0° and 359°) from the compass data, we obtain the desired compass value, still viewing in northern direction after rotating the device (cf. Fig. 5).

3.1.2 Point of Interest Controller

As soon as the *Sensor Controller* has collected the required data, it notifies the *Point of Interest Controller* at two points in time: (1) when detecting a new location and (2) after having gathered new heading as

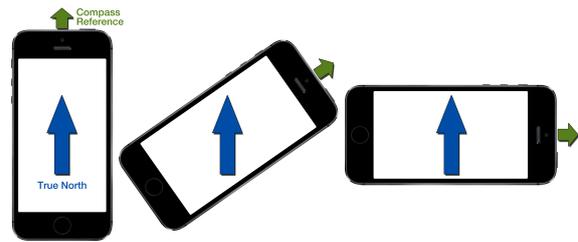


Figure 5: Adjusting the compass data to the device's current rotation.

well as accelerometer data. When a new location is detected, we must determine the POIs in the surrounding of the user. For this purpose, we use an adjustable radius (see Fig. 9 for an example of such an adjustable radius). By using the latter, a user can determine the maximum distance she has to the POIs to be displayed. By calculating the distance between the device and the POIs based on their GPS coordinates (Bullock, 2007), we can determine the POIs located inside the chosen radius and hence the POIs to be displayed on the screen. Since only POIs inside the field of view (i.e., POIs actually visible for the user) shall be displayed on the screen, we must further calculate the vertical and horizontal bearing of the POIs inside the radius. Due to space limitation, we cannot describe these calculations in detail, but refer interested readers to a technical report (Geiger et al., 2013). As explained in this report, the vertical bearing can be calculated based on the altitudes of the POIs and the smart mobile device (the latter can be determined from the current GPS coordinates). In turn, the horizontal bearing can be computed using the *Haversine* formula (Sinnott, 1984) and applying it to the GPS coordinates of the POI and the smart mobile device. Finally, in order to avoid recalculations of these surrounding POIs in case the GPS coordinates do not change (i.e., within movings of 10m), we must buffer data of the POIs inside the controller implementation for efficiency reasons.

As a next step, the heading and accelerometer data need to be processed when obtaining a notification from the *Sensor Controller* (i.e., the application loop mentioned in Section 3.1.1 has delivered new data). Based on this, we can determine whether or not a POI is located inside the vertical and horizontal field of view, and at which position it shall be displayed on the *locationView*. Recall that the *locationView* extends the actual field of view to a larger, orientation-independent field of view (cf. Fig. 6). The first step is to determine the boundaries of the *locationView* based on sensor data. In this context, the heading data provides the information required to determine the direction the device is pointing at. The left boundary of the *locationView* can be calculated by determining

the horizontal heading and decreasing it by the half of the maximal angle of view (cf. Fig. 6). The right boundary is calculated by adding half of the maximal angle of view to the current heading. Since POIs have also a vertical heading, a vertical field of view must be calculated as well. This is done analogously to the calculation of the horizontal field of view, except that the data of the vertical heading is required. Finally, we obtain a directed, orientation-independent field of view bounded by left, right, top, and bottom values. Then we use the vertical and horizontal bearings of a POI to determine whether it lies inside the *locationView* (i.e., inside the field of view). Since we use the concept of the *locationView*, we do not have to deal with the rotation of the device at this point, i.e., we can normalize calculations to portrait mode since the rotation itself is handled by the *locationView*.

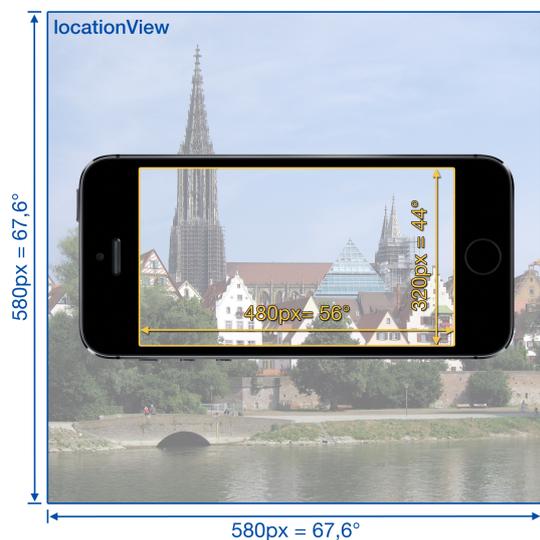


Figure 6: Illustration of the new maximal angle view and the real one.

The camera view can be created and displayed applying the native *AVFoundation* framework. Using the screen size of the device, which can be determined at run time, the *locationView* can be initialized and placed centrally on top of the camera view. As soon as the *Point of Interest Controller* has finished its calculations (i.e., it has determined the positions of the POIs), it notifies the *View Controller* that organizes the view components. The *View Controller* then receives the POIs and places them on the *locationView*. Recall that in case of a device rotation, only the *locationView* must be rotated. As a consequence, the actual visible field of view changes accordingly. Therefore, the *Point of Interest Controller* sends the rotation of the device calculated by the *Sensor Controller* to the *View Controller*, together with the POIs. Thus, we

can adjust the field of view by simply counterrotating the *locationView* using the given angle. Based on this, the user will only see those POIs on his screen, being inside the actual field of view. In turn, other POIs will be hidden after the rotation, i.e., moved out of the screen (cf. Fig. 1). Detailed insights into respective implementation issues, together with well described code samples, can be found in (Geiger et al., 2013).

3.2 Android Mobile Operating System

In general, mobile business applications should be made available on all established platforms in order to reach a large number of users. Hence, we developed AREA for the Android mobile operating system as well (and will also make it available for Windows Phone at a later point in time). This section gives insights into the Android implementation of AREA, comparing it with the corresponding iOS implementation. Although the basic software architecture of AREA is the same for both mobile operating systems, there are differences regarding its implementation.

3.2.1 Sensor Controller

For implementing the *Sensor Controller*, the packages *android.location* and *android.hardware* are used. The *location package* provides functions to retrieve the current GPS coordinate and altitude of the respective device, and is similar to the corresponding iOS package. However, the Android location package additionally allows retrieving an approximate position of the device based on network triangulation. Particularly, if no GPS signal is available, the latter approach can be applied. However, as a drawback, no information about the current altitude of the device can be determined in this case. In turn, the *hardware package* provides functions to get notified about the current magnetic field and accelerometer. The latter corresponds to the one of iOS, and is used to calculate the rotation of the device. However, the heading is calculated in a different way compared to iOS. Instead of obtaining it with the location service, it must be determined manually. Generally, the heading depends on the rotation of the device and the magnetic field. Therefore, we create a rotation matrix using the data of the magnetic field (i.e., a vector with three dimensions) and the rotation based on the accelerometer data. Since the heading data depends on the accelerometer as well as the magnetic field, it is rather inaccurate. More precisely, the calculated heading is strongly oscillating. Hence, we apply a lowpass filter to mitigate this oscillation. Note that this lowpass filter is of another type than the one used in Section 3.1.1 for calculating the gravity.

Moreover, as soon as other magnetic devices are located nearby the actual mobile device, the heading will be distorted. In order to notify the user about the presence of such a disturbed magnetic field, leading to false heading values, we apply functions of the hardware package. Another difference between iOS and Android concerns the way the required data can be obtained. Regarding iOS, location-based data is pushed, whereas sensor data must be polled. As opposed to iOS, on Android all data is pushed by the framework, i.e., application programmers rely on Android internal loops and trust the up-to-dateness of the data provided. Note that such subtle differences between mobile operating systems and their development frameworks should be well understood by the developers of advanced mobile business applications.

3.2.2 Point of Interest Controller

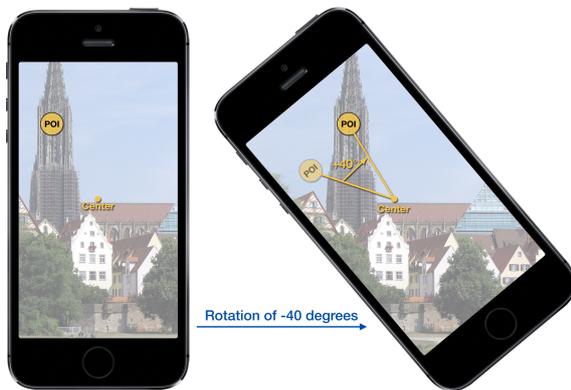


Figure 7: Android specific rotation of POI and field of view.

Regarding Android, the *Point of Interest Controller* works the same way as the one of iOS. However, when developing AREA we had to deal with one particular issue. The *locationView* manages the visible POIs as described above. Therefore, it must be able to add child views (e.g., every POI generating one child view). As described in Section 3.1, on iOS we simply rotate the *locationView* to actually rotate the POIs and the field of view. In turn, on Android, a layout containing child views cannot be rotated in the same way. Thus, when the *Point of Interest Controller* receives sensor data from the *Sensor Controller*, the x- and y-coordinates of the POIs must be determined in a different way. Instead of placing the POIs independently of the current rotation of the device, we make use of the degree of rotation provided by the *Sensor Controller*. Following this, the POIs are rotated around the centre of the *locationView* and we also rotate the POIs about their centres (cf. Fig. 7). Using this approach, we can still add all POIs to the field of view



Figure 8: AREA's user interface for iOS and Android.

of the *locationView*. Finally, when rotating the POIs, they will automatically leave the device's actual field of view.

3.3 Comparison

This section compares the two implementations of AREA on iOS and Android. First of all, it is noteworthy that the features and functions of the two implementations are the same. Moreover, the user interfaces realized for AREA on iOS and Android, respectively, are the same (see Fig. 8).

3.3.1 Realizing the LocationView

The developed *locationView* with its specific features differs between the Android and iOS implementations of AREA. Regarding the iOS implementation, we could realize the *locationView* concept as described in Section 2.1. On the Android operating system, however, not all features of this concept worked properly. More precisely, extending the actual field of view of the device to the bigger size of the *locationView* worked well. Furthermore, determining whether or not a POI is inside the field of view, independent of the rotation of the device, worked also well. By contrast, rotating the *locationView* with its POIs to adjust the visible field of view as well as moving invisible POIs out of the screen did not work as easy on Android as expected. As particular problem in the given context, a simple view on Android must not contain any child views. Therefore, on Android we had to use the *layout* concept for realizing the described *locationView*. However, simply rotating a layout does not work on all Android devices. For example, on a Nexus 4 device this worked well by implementing the algorithm in exactly the same way as on iOS. In

turn, on a Nexus 5 device this led to failures regarding the redraw process. When rotating the layout on the Nexus 5, the *locationView* is clipped by the camera surface view, which is located behind our *locationView*. As a consequence, to ensure that AREA is compatible with a wider set of Android devices, running Android 4.0 or later, we made the adjustments described in Section 4.2.

3.3.2 Accessing Sensors

Using sensors on the two mobile operating systems is different as well. The latter concerns the access to the sensors as well as their preciseness and reliability. Regarding iOS, the location sensor is offering the GPS coordinates as well as the compass heading. This data is pushed to the application by the underlying service offered by iOS. Concerning Android, the location sensor only provides data of the current location. Furthermore, this data must be polled by the application. The heading data, in turn, is calculated by the fusion of several motion sensors, including the accelerometer and magnetometer. The accelerometer is used on both platforms to determine the current orientation of the device. However, the preciseness of data provided differs significantly. Running and compiling the AREA engine on iOS with iOS 6 results in very reliable compass data with an interval of one degree. Running and compiling the AREA engine with iOS 7, however, leads to different results compared to iOS 6. As advantage, iOS 7 enables a higher resolution of the data intervals provided by the framework due to the use of floating point data instead of integers. In turn, the partial unreliability of the delivered compass data is disadvantageous. Regarding iOS 7, compass data started to oscillate within an interval when moving the device. Therefore, we needed to apply a stronger low-pass filter in order to compensate this oscillating data. In turn, on Android the internal magnetometer, which is necessary for calculating the heading, is vulnerable to noisy sources (e.g., other devices, magnets, or computers). Thus, it might happen that the delivered data is unreliable and the application must wait until more reliable sensor data becomes available.

Furthermore, for each sensor the corresponding documentation on the respective operating system should be studied in detail in order to operate with them efficiently. In particular, the high number of different devices running Android constitutes a challenge when deploying AREA on the various hardware and software configurations of manufacturers. Finally, we learned that Android devices are often affected by distortions of other electronic hardware and, therefore, the delivered data might be unreliable as well.

Overall, the described differences demonstrate that developing advanced mobile business applications, which make use of the technical capabilities of modern smart mobile devices, is far from being trivial from the viewpoint of application developers.

4 VALIDATION

This section deals with the development of business applications with AREA and the lessons learned in this context.

4.1 Developing Business Applications with AREA

AREA has been integrated with several business applications. For example, one company uses AREA for its application *LiveGuide* (CMCityMedia, 2013). A *LiveGuide* can be used to provide residents and tourists of a German city with the opportunity to explore their surrounding by displaying points of interests stored for that city (e.g., public buildings, parks, places of events, or companies). When realizing such business applications on top of AREA, it turned out that their implementation benefits from the modular design and extensibility of AREA. Furthermore, an efficient implementation could be realized. In particular, when developing the *LiveGuide* application type, only the following two steps were required: First, the appearance of the POIs was adapted to meet the user interface requirements of the respective customers. Second, the data model of AREA was adapted to an already existing one. On the left side of Fig. 9, we show user interface elements we made in the context of the *LiveGuide* applications. In turn, on the right side of Fig. 9, we show the user interface elements originally implemented for AREA.

4.2 Lessons Learned

This section discusses issues that emerged when developing business applications (e.g., *LiveGuide*) on top of AREA. Note that we got many other practical insights from the use of AREA. However, to set a focus, we restrict ourselves to two selected issues.

4.2.1 Updates of Mobile Operating Systems

As known, the iOS and Android mobile operating systems are frequently updated. In turn, respective updates must be carefully considered when developing and deploying an advanced mobile business application like AREA. Since the latter depends on

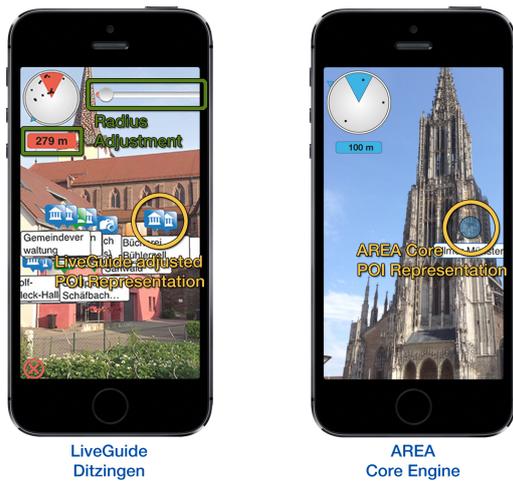


Figure 9: Typical adapted user interface provided by a LiveGuide application.

| | |
|--|--|
| <pre>public void onSensorChanged(SensorEvent event) { if (event.accuracy == SensorManager.SENSOR_STATUS_UNRELIABLE) { // is called when read data is inaccurate notifyAccuracyChanged(event.accuracy); } ... }</pre> | <pre>public void onAccuracyChanged(Sensor s, int accur) { // is called when read data's accuracy has changed notifyAccuracyChanged(accur); }</pre> |
| Android 4.2 | Android 4.3 |

Figure 10: SENSOR_STATUS_UNRELIABLE change in Android 4.3.

the availability of accurate sensor data, fundamental changes of the respective native libraries might affect the proper execution of AREA. As example, consider the following issue we had to cope with in the context of an update of the Android operating system (i.e., the switch from Android Version 4.2 to Version 4.3). In the old version, the sensor framework notifies AREA when measured data becomes unreliable. However, with the new version of the mobile operating system, certain constants (e.g., *SENSOR_STATUS_UNRELIABLE*) we had used were no longer known on respective devices (cf. Fig. 10). To deal with this issue, the respective constant had to be replaced by a listener (cf. Fig. 10 *onAccuracyChanged*). As another example consider the release of iOS 7, which led to a change of the look and feel of the entire user interface. In particular, some of the customized user interface elements in the deployed version of the *LiveGuide* applications got hidden from one moment to the other or did not react to user interactions anymore. Thus, the application had to be fixed. Altogether, we learned that adjusting mobile applications due to operating system updates might cause considerable efforts.

4.2.2 POI Data Format

Using our own proprietary XML schema instead of applying and adapting the open source schema ARML has its pros and cons. On one hand, we can simply extend and modify this schema, e.g., to address upcoming issues in future work. On the other, when integrating AREA with the *LiveGuide* application, we also revealed drawbacks of our approach. In particular, the data format of POIs, stored in external databases, differs due to the use of a non-standardized format. Thus, the idea of ARML (ARML, 2013) is promising. Using such a standardized format for representing POIs from different sources should be pursued. Therefore, we will adapt AREA to support this standard with the goal to allow for an easy integration of AREA with other business applications.

5 RELATED WORK

Previous research related to the development of a location-based augmented reality application, which is based on GPS coordinates and sensors running on *head-mounted* displays, is described in (Feiner et al., 1997) and (Koopman and MacIntyre, 2003). In turn, a simple smart mobile device, extended by additional sensors, has been applied by (Kähäri and Murphy, 2006) to develop an augmented reality system. Another application using augmented reality is described in (Lee et al., 2009). Its purpose is to share media data and other information in a real-world environment and to allow users to interact with this data through augmented reality. However, none of these approaches addresses location-based augmented reality on smart mobile devices as AREA does. In particular, these approaches do not give insights into the development of such business applications.

The increasing size of the smart mobile device market as well as the technical maturity of smart mobile devices has motivated software vendors to realize *augmented reality software development kits* (SDKs). Example of such SDKs included *Wikitude* (Wikitude, 2013), *Layar* (Layar, 2013), and *Junaio* (Junaio, 2013). Besides these SDKs, there are popular applications like *Yelp* (Yelp, 2013), which use additional features of augmented reality to assist users when interacting with their surrounding.

Only little work can be found, which deals with the development of augmented reality systems in general. As an exception, (Grubert et al., 2011) validates existing augmented reality browsers. However, neither commercial software vendors nor scientific results related to augmented reality provide any insight

into how to develop a location-based mobile augmented reality engine.

6 SUMMARY & OUTLOOK

The purpose of this paper was to give insights into the development of the core framework of an augmented reality engine for smart mobile devices. We have further shown how business applications can be implemented based on the functionality of this mobile engine. As demonstrated along selected implementation issues, such a development is very challenging. First of all, a basic knowledge about mathematical calculations is required, i.e., formulas to calculate the distance and heading of points of interest on a sphere in the context of outdoor scenarios. Furthermore, deep knowledge about the various sensors of the smart mobile device is required from application developers, particularly regarding the way the data provided by these sensors can be accessed and processed. Another important issue concerns resource and energy consumption. Since smart mobile devices have limited resources and performance capabilities, the points of interest should be displayed in an efficient way and without delay. Therefore, the calculations required to handle sensor data and to realize the general screen drawing that must be implemented as efficient as possible. The latter has been accomplished through the concept of the *locationView*, which allows increasing the field of view and reusing already drawn points of interest. In particular, the increased size allows the AREA engine to easily determine whether or not a point of view is inside the *locationView* without considering the current rotation of the smart mobile device. In addition, all displayed points of interest can be rotated easily.

We argue that an augmented reality engine like AREA must provide a sufficient degree of modularity to enable a full and easy integration with existing applications as well as to implement new applications on top of it. Finally, it is crucial to realize a proper architecture and class design, not neglecting the communication between the components. We have further demonstrated how to integrate AREA in a real-world business applications (i.e., *LiveGuide*) and how to make use of AREA's functionality. In this context, the respective application has been made available in the Apple App and Android Google Play Stores. In particular, the realized application has shown high robustness. Finally, we have given insights into the differences between Apple's and Google's mobile operating systems when developing AREA.

Future research on AREA will address the chal-

lenges we identified during the implementation of the *LiveGuide* business application. For example, in certain scenarios the POIs located in the same direction overlap each other, making it difficult for users to precisely touch POIs. To deal with this issue, we are working on algorithms for detecting clusters of POIs and offering a way for users to interact with these clusters. In (Feineis, 2013), a component for on-the-trail navigation in mountainous regions has been developed on top of AREA, which is subject of current research as well. Furthermore, we are developing a *marker-based* augmented reality component in order to integrate marker based with location based augmented reality. Since GPS is only available for outdoor location, but AREA should also for indoor scenarios, we are working towards this direction as well. In the latter context, we use Wi-Fi triangulation to determine the device's indoor position (Bachmeier, 2013). Second, we are experiencing with the iBeacons approach introduced by Apple.

Finally, research on business process management offers flexible concepts, which are useful for enabling proper exception handling in the context of mobile applications as well (Pryss et al., 2012; Pryss et al., 2013; Pryss et al., 2010). Since mobile augmented reality applications may cause various errors (e.g., sensor data is missing), adopting these concepts is promising.

REFERENCES

- Alasdair, A. (2011). *Basic Sensors in iOS: Programming the Accelerometer, Gyroscope, and More*. O'Reilly Media.
- Apple (2013). Event handling guide for iOS: Motion events. [Online; accessed 10.12.2013].
- ARML (2013). Augmented reality markup language. <http://openarml.org/wikitude4.html>. [Online; accessed 10.12.2013].
- Bachmeier, A. (2013). Wi-fi based indoor navigation in the context of mobile services. *Master Thesis, University of Ulm*.
- Bullock, R. (2007). Great circle distances and bearings between two locations. [Online; accessed 10.12.2013].
- Carmigniani, J., Furht, B., Anisetti, M., Ceravolo, P., Damiani, E., and Ivkovic, M. (2011). Augmented reality technologies, systems and applications. *Multimedia Tools and Applications*, 51(1):341–377.
- CMCityMedia (2013). City liveguide. <http://liveguide.de>. [Online; accessed 10.12.2013].
- Corral, L., Sillitti, A., and Succi, G. (2012). Mobile multiplatform development: An experiment for performance analysis. *Procedia Computer Science*, 10(0):736 – 743.

- Feineis, L. (2013). Development of an augmented reality component for on the trail navigation in mountainous regions. *Master Thesis, University of Ulm, Germany*.
- Feiner, S., MacIntyre, B., Höllerer, T., and Webster, A. (1997). A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. *Personal Technologies*, 1(4):208–217.
- Fröhlich, P., Simon, R., Baillie, L., and Anegg, H. (2006). Comparing conceptual designs for mobile access to geo-spatial information. *Proc of the 8th Conf on Human-computer Interaction with Mobile Devices and Services*, pages 109–112.
- Geiger, P., Pryss, R., Schickler, M., and Reichert, M. (2013). Engineering an advanced location-based augmented reality engine for smart mobile devices. Technical Report UIB-2013-09, University of Ulm, Germany.
- Grubert, J., Langlotz, T., and Grasset, R. (2011). Augmented reality browser survey. *Technical report, Institute for Computer Graphics and Vision, Graz University of Technology, Austria*.
- Junaio (2013). Junaio. <http://www.junaio.com/>. [Online; accessed 11.06.2013].
- Kähäri, M. and Murphy, D. (2006). Mara: Sensor based augmented reality system for mobile imaging device. *5th IEEE and ACM Int'l Symposium on Mixed and Augmented Reality*.
- Kamenetsky, M. (2013). Filtered audio demo. http://www.stanford.edu/~boyd/ee102/conv_demo.pdf. [Online; accessed 17.01.2013].
- Kooper, R. and MacIntyre, B. (2003). Browsing the real-world wide web: Maintaining awareness of virtual information in an AR information space. *Int'l Journal of Human-Computer Interaction*, 16(3):425–446.
- Layar (2013). Layar. <http://www.layar.com/>. [Online; accessed 11.06.2013].
- Lee, R., Kitayama, D., Kwon, Y., and Sumiya, K. (2009). Interoperable augmented web browsing for exploring virtual media in real space. *Proc of the 2nd Int'l Workshop on Location and the Web*, page 7.
- Paucher, R. and Turk, M. (2010). Location-based augmented reality on mobile phones. *IEEE Computer Society Conf on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 9–16.
- Pryss, R., Langer, D., Reichert, M., and Hallerbach, A. (2012). Mobile task management for medical ward rounds - the MEDo approach. *Proc BPM'12 Workshops*, 132:43–54.
- Pryss, R., Musiol, S., and Reichert, M. (2013). Collaboration support through mobile processes and entailment constraints. *Proc 9th IEEE Int'l Conf on Collaborative Computing (CollaborateCom'13)*.
- Pryss, R., Tiedeken, J., Kreher, U., and Reichert, M. (2010). Towards flexible process support on mobile devices. *Proc CAiSE'10 Forum - Information Systems Evolution*, (72):150–165.
- Reitmayr, G. and Schmalstieg, D. (2003). Location based applications for mobile augmented reality. *Proc of the Fourth Australasian user interface conference on User interfaces*, pages 65–73.
- Robecke, A., Pryss, R., and Reichert, M. (2011). Dbisolar: An iphone application for performing citation analyses. *Proc CAiSE'11 Forum at the 23rd Int'l Conf on Advanced Information Systems Engineering*, (Vol-73).
- Schobel, J., Schickler, M., Pryss, R., Nienhaus, H., and Reichert, M. (2013). Using vital sensors in mobile healthcare business applications: Challenges, examples, lessons learned. *Int'l Conf on Web Information Systems and Technologies*, pages 509–518.
- Sinnott, R. (1984). Virtues of the haversine. *Sky and telescope*, 68:2:158.
- Systems, A. (2013). Phonegap. <http://phonegap.com>. [Online; accessed 10.12.2013].
- Wikitude (2013). Wikitude. <http://www.wikitude.com>. [Online; accessed 11.06.2013].
- Yelp (2013). Yelp. <http://www.yelp.com>. [Online; accessed 11.06.2013].

SHORT PAPER

Alternative Communication System for Emergency Situations

I. Santos-González, A. Rivero-García, P. Caballero-Gil and C. Hernández-Goya

*Department of Computing, University of La Laguna, Tenerife, Spain
ivan.santos.99@ull.edu.es, alelit4@gmail.com, {pcaballe, mchgoya}@ull.es*

Keywords: Wi-Fi Direct, Security, Mobile Communications, Android, SNOW 3G.

Abstract: Nowadays, many different technologies can be used for wireless communication in various types of situations. However, most of them depend on network infrastructures like mobile communication antennas, which can be unavailable due to distinct problems such as natural disasters, terrorist attacks or huge people agglomerations in massive events like demonstrations, concerts or sport events. This paper presents a solution to this problem, which is based on a new communication system that uses the Wi-Fi Direct technology to deploy secure communications independently of the network infrastructure.

1 INTRODUCTION

In the last years, many mobile devices like smartphones, laptops, printers, personal computers, tablet PCs, etc. have included among their technical features a new wireless technology called Wi-Fi Direct, which is a standard that allows Wi-Fi devices to establish a connection with no need of an access point. This technology manages that Wi-Fi Direct devices can transfer data between them directly, with a configuration that is much easier than the one required doing it with the traditional Wi-Fi technology.

The variety of advantages that Wi-Fi Direct technology offers, and the fact that it does not require any additional infrastructure, make that this technology can be very useful in emergency situations. Due to this, and to the fact that very often the usual infrastructures can be unavailable in this kind of situations; this paper proposes a new wireless communication system based on this technology.

Furthermore, since new smartphones include in their technical features the Wi-Fi Direct technology, the use of the proposed wireless communication system based on this technology is possible simply by using devices that are very common nowadays.

The proposed communication system pays special attention to the deployment of secure communications in emergency or natural disaster situations by using different cryptographic algorithms and protocols. Moreover, the described system allows users to send automatically their

geographic coordinates in each message if they wish. This feature makes it possible that emergency services can provide evacuation and rescues in a more efficient way.

The described system is already available to users through the development and publication of an Android application that allows them to deploy secure communications automatically over Wi-Fi Direct technology.

This work is organized as follows. Section 2 gives a brief description of Wi-Fi Direct technology. Afterwards, Section 3 introduces several concepts related to security, which are necessary for the definition of the mobile application called Wi-Fi Direct Locator, described in Section 4. Finally, Section 5 closes the paper with some conclusions and open questions

2 WI-FI DIRECT TECHNOLOGY

Wi-Fi Direct is a new wireless technology developed by the Wi-Fi Alliance with the goal of establishing Peer TO Peer (P2P) communications between users using the Wi-Fi technology (Wi-Fi Alliance, 2013).

Traditionally, P2P systems can be deployed in two different ways.

On the one hand, the so-called ad hoc mode consists of two devices that are interconnected, forming a point-to-point network. In this mode, each device plays the roles of client and access point simultaneously. This configuration is known as Independent Basic Service Set (IBSS) (Ilyas, 2010)

and corresponds to a connection where each device acts like a chain node, broadcasting the information to the next node.

On the other hand, it is possible to exchange packets between two devices using an Access Point to broadcast the information to the receiver. This mode that try to imitate the P2P connection is called Tunnelled Direct Link Setup (TDLS) (Rajamani, Wentink, Jones, 2011).

These two modes have not yet been used due to the absence of standardized Medium Access Control (MAC) protocols to broadcast packets over WLAN ad hoc networks. Due to these problems, it was necessary to create a new standard to make the task of connecting two devices over Wi-Fi technology easier. That was the main goal of Wi-Fi Direct.

Wi-Fi Direct extends the Wi-Fi technology adding new capabilities (Wi-Fi Alliance, 2010). One of its main features consists on the possible configuration of groups where a device, called Group Owner, is able to establish multiple P2P connections with different devices, called Group Clients. Consequently, the Group Owner works as an access point in the Wi-Fi infrastructure mode and the other devices join the group as clients in traditional Wi-Fi, with the added ability of establishing a unique P2P connection with the device playing the role of Group Owner.

The main advantage of Wi-Fi Direct compared with other similar technologies is that all necessary changes with respect to the traditional Wi-Fi are done at software level. This fact assures backward compatibility with all certified Wi-Fi devices, so this technology is being widespread easily.

In Table 1, we can see the Wi-Fi Direct theoretical specifications.

Table 1: Wi-Fi Direct specifications.

| <i>Specification</i> | <i>Value</i> |
|----------------------|----------------------------------|
| Bandwidth | 250 Mbps |
| Range | 100 m |
| Frequency | 2.4 GHz |
| Energy | 25 % less than traditional Wi-Fi |
| Security | WPA2 |

3 SECURITY CONCEPTS

In Wi-Fi Direct, basic security is provided by

WPA2, which is the commercial name of the final version of the 802.11i standard adopted by the Wi-Fi Alliance in June 2004.

WPA2 improves the security of WPA, forcing the use of a stronger cipher. WPA2 uses AES algorithm cipher, which is one of the most secure encryption algorithms nowadays. Furthermore, it does not allow the use of the TKIP (Temporal Key Integrity Protocol) algorithm in order to avoid some security holes that are known in this algorithm (Halvorsen et al, 2009).

Wi-Fi Direct uses the WPA2-Personal version of WPA2, works incorporating a limited access point in each device, and uses the WPS (Wi-Fi Protected Setup) system to negotiate each link. In this way, Wi-Fi Direct devices can work either like a client or like an access point, so the first time that two devices establish a connection, both negotiate and determine which will be the access point.

The WPS system is a standard defined to create secure Wi-Fi networks, including some mechanisms to facilitate the configuration of a Wi-Fi network using WPA2. It is thought to minimize the user intervention in small offices and domestic environments.

Wi-Fi network security in general has been questioned for a long time due to several practical attacks that have been launched against the successive versions of the technology: WEP, WPA and WPA2. Particularly, around a year ago a security breach in Wi-Fi protected setup was published that affects wireless routers that use WPS (Viehböck, 2011). This failure allows attackers to obtain the PIN (Personal Identification Number) and the Pre-Shared Key (PSK) in a WPA/WPA2 network in a few hours. In order to protect the network, users should disable the WPS functions in their network as a temporal solution. However, this solution is not possible in all devices.

Due to the aforementioned and other security problems of Wi-Fi networks and, by extension, of Wi-Fi Direct, this work proposes an additional security scheme for Wi-Fi Direct networks, based on a combination of the Elliptic Curve Diffie-Hellman protocol and the SNOW 3G algorithm.

3.1 Elliptic Curve Cryptography

The use of elliptic curves in cryptography is due to the fact that they provide a security level equivalent to the one of other systems, but using a much shorter key length. This property implies that when using elliptic curves cryptosystems, we use a lesser amount of memory and a lower bandwidth

consumption to save and to transmit the key respectively.

In cryptography, elliptic curves defined over finite fields $K = Z_p$, being p a prime number, offer several interesting features.

A finite field is a finite set of elements with the addition and multiplication operations satisfying the associative, commutative and distributive properties, and having an additive inverse and a neutral element both for the sum and for the multiplication (Blake, Seroussi, Smart, 1999).

In elliptic curves defined over finite fields of high dimensions, the computation of a multiple of a point is very difficult. This is known as the Elliptic Curve Discrete Logarithm Problem and it is a problem similar to that of computing the discrete logarithm in a finite multiplicative group.

The security of cryptographic procedures based on elliptic curves depends on the complexity of the calculation of the elliptic logarithm. Such a complexity, when using non-supersingular curves, what is usual in cryptography, is \sqrt{p} . This value is much greater than the complexity of the factorization problem or the discrete logarithm problem (Smart, 1999).

Some of the properties of the elliptic curves defined over finite fields allow us to say that when we work with integer numbers sufficiently large, the use of the elliptic logarithm can be more secure than the use of factorization or of discrete logarithm. This fact is shown in Table 2.

Table 2: Elliptic curve security.

| Key length | Cryptosystem | Operations (cipher) | Break Time (PC 109 FLOPS) |
|------------|-----------------|---------------------|---------------------------|
| 30 | RSA | $9,0 * 10^2$ | 0.3 seconds |
| 30 | Elliptic curves | $2,7 * 10^4$ | 11 days |
| 50 | RSA | $2,5 * 10^3$ | 2 minutes |
| 50 | Elliptic curves | $1,2 * 10^5$ | $3,0 * 10^3$ years |
| 100 | RSA | $1,0 * 10^4$ | 28 days |
| 100 | Elliptic curves | $1,0 * 10^5$ | --- |
| 200 | RSA | $4,0 * 10^4$ | $3,8 * 10^6$ years |
| 200 | Elliptic curves | $8,0 * 10^5$ | --- |

3.2 Elliptic Curve Diffie-Hellman Protocol

The Elliptic Curve Diffie-Hellman (ECDH) Protocol

is a variation of the original Diffie-Hellman protocol, which uses the properties of the elliptic curves defined over finite fields (Koblitz, 1987) (Miller, 1985).

In this way, the two users agree beforehand on the use of a prime number p , an elliptic curve E defined over Z_p and a point $P \in E$.

Then, the users A and B choose as secret keys two random numbers belonging to Z_p , being these $a \in Z_p$ and $b \in Z_p$.

Later, they obtain their public keys multiplying their secret keys by the point P previously agreed.

The next step is the exchange of their public keys in order to compute later the shared key by multiplying their private key by the public key of the other user, obtaining both the same shared key.

The strength of this protocol lies on the difficulty to solve the Elliptic Curve Discrete Logarithm Problem.

However, since this protocol derives from the original Diffie-Hellman protocol, it is susceptible to suffer the same attacks.

3.3 SNOW 3G Algorithm

SNOW 3G is the stream cipher algorithm designated in 2006 as base for the integrity protection and encryption of the UMTS technology.

Thanks to the fact that the algorithm satisfies all the requirements imposed by the 3GPP (3rd Generation Partnership Project) with respect to time and memory resources, it was selected as base of the UMTS Encryption Algorithm 2 (UEA2) and UMTS Integrity Algorithm 2 (UIA2) algorithms (IA UEA2&UIA, 2006).

The SNOW 3G algorithm derives from the SNOW 2 algorithm, and uses 128-bit keys and an initialization vector in order to generate in each iteration 32 bits of keystream.

The LFSR used in this algorithm has 16 stages denoted $s_0, s_1, s_2 \dots s_{15}$ with 32 bits each one.

On the other hand, the used FSM (Finite State Machine) is based on three 32-bit records denoted $R1, R2$ and $R3$ and uses two Substitution-boxes called $S1$ and $S2$.

The combination operation uses a XOR and an addition module 2^{32} .

Figure 1 shows the general scheme of this algorithm.

This algorithm has two execution modes: the initialization mode and the keystream mode. First, the initialization mode is executed without producing any keystream. Then, the keystream mode is executed. In particular, the number of iterations of

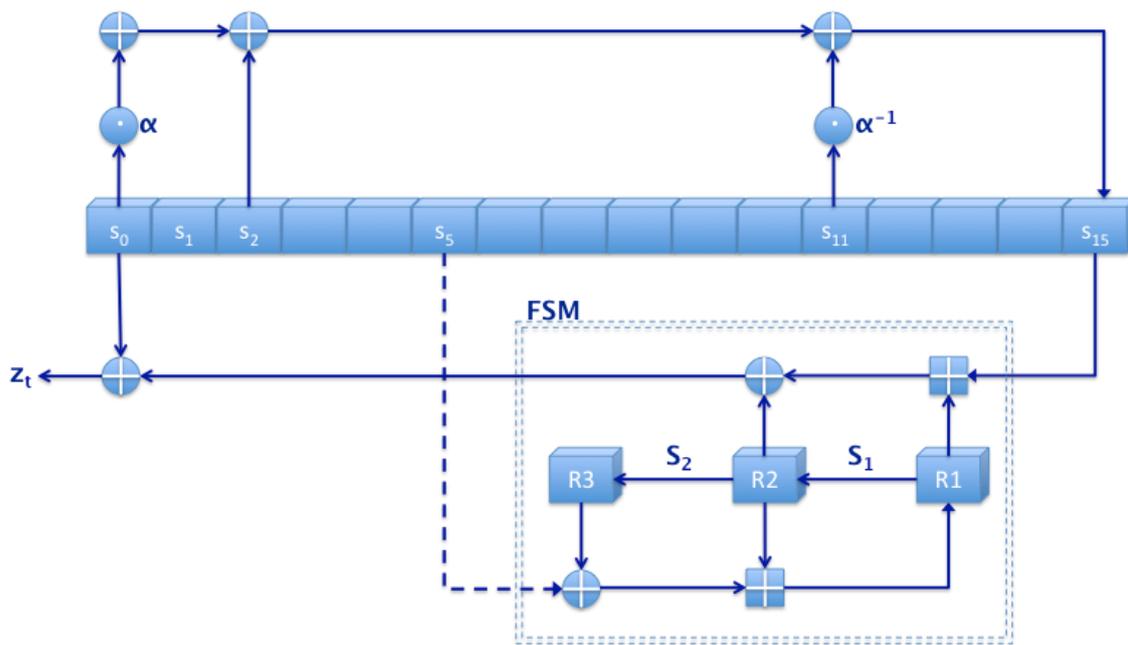


Figure 1: SNOW 3G general scheme.

such a mode depends of the number of 32-bit words that we want to generate.

4 WI-FI DIRECT LOCATOR APPLICATION

The new communication system that is here presented uses the Wi-Fi Direct technology and the aforementioned security algorithms, and it is called Wi-Fi Direct Locator. It is an Android application that has as its main purpose the establishment of secure communication by means of Wi-Fi Direct technology, to be used in catastrophic and natural disaster situations where the network infrastructures are not available.

With Wi-Fi Direct Locator we can exchange text messages between users in a safe and free way. This mobile application uses the ECDH protocol and the SNOW 3G stream cipher. It is remarkable that it is the first Android application that uses the SNOW 3G algorithm. Simultaneously, a java library that implements the SNOW 3G stream cipher has been developed for the application, and it is also the first java library that implements this algorithm.

Furthermore, Wi-Fi Direct Locator application can localize the geographic coordinates of the origin of any sent message if the users wish to. In order to see the corresponding point on the map, users only have to press over the received or sent message and

a map will appear with the point marked on it.

Wi-Fi Direct Locator has an automatic mode that works sending periodically location messages according to a period of time defined by the user. This mode is especially interesting in emergency situations when a person cannot use the mobile phone because he/she is injured or trapped, because in this way the rescue services or other people will be able to rescue him/her faster.

This application uses the Android ge-positioning Application Programming Interface (API) that allows obtaining location information through Global Positioning System (GPS).

The use of this API in conjunction with the Google Maps API provides the application the ability to show our position in Google Maps and interact with it in different ways.

We have decided to implement the messaging system between users using transmission Control Protocol (TCP) sockets that allows us to forget the retransmission of shipping being the TCP protocol the responsible of forward the packets that do not arrive to their destination.

The sockets work in two different ways depending on the fact that they are server sockets or client sockets. The server sockets wait a client to establish a connection while the client sockets search a server socket to establish it.

Wi-Fi Direct Locator interface has been designed following the principles purpose by Google in its design section.

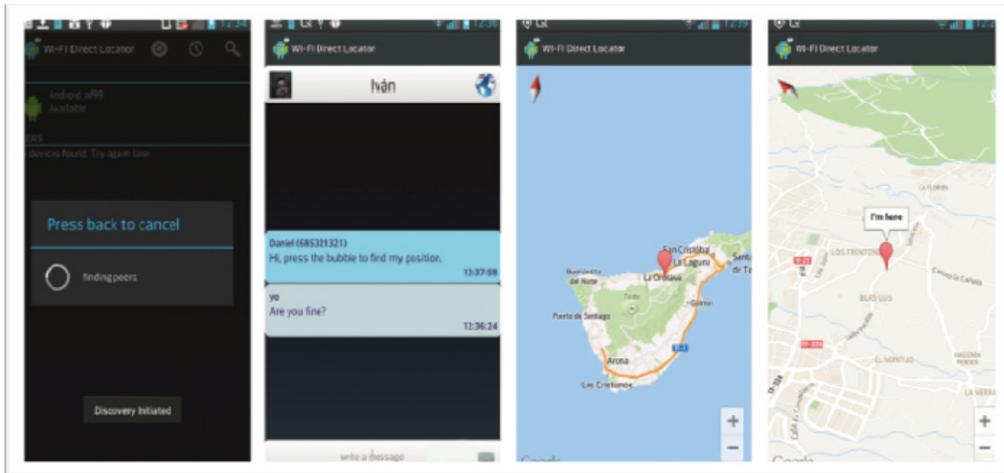


Figure 2: Screenshots of Wi-Fi Direct Locator.

During the application development we have focused on some specific details to improve the user's experience.

First, we have reduced the number of navigation screens to do the user's navigation easier and we use images buttons instead of traditional buttons to capture the user's attention.

We have also endeavoured to do our application similar to the system application look. For this reason, Android defines three specific themes, and we use one of these (Holo Dark) to do our application visually similar to the predetermined system applications.

Wi-Fi Direct Locator is available in Google Play market and it has been downloaded by many people in different countries to communicate in emergency situations without the need of network infrastructures.

Figure 2 shows the user's interface by means of a series of images of different navigation screens.

5 CONCLUSIONS

This paper describes a new communication system based on the Wi-Fi Direct technology to deploy secure communications that are useful in situations when network infrastructures are unavailable. Thus, the proposal is especially useful under conditions such as natural disasters, terrorist attacks or simply huge agglomerations of people in massive events like demonstrations, concerts or sport events. Among the main features of the proposal, security is the most remarkable because cryptographic algorithms based on elliptic curve

cryptography and stream ciphers are used.

The proposed system is already available to users through an Android application. However, there are still many open problems such as the study of attacks resistance and a wider automation of the application.

ACKNOWLEDGEMENTS

Research supported by the Spanish MINECO and the European FEDER Funds under projects TIN2011-25452 and IPT-2012-0585-370000.

REFERENCES

- Blake, I. F., Seroussi, G., Smart, N., 1999. *Elliptic curves in cryptography*, Vol. 265. Cambridge university press.
- Halvorsen, F.M., Haugen, O., Eian, M., Mjøl̄snes, S.F., 2009. An improved attack on TKIP. *Identity and Privacy in the Internet Age*, pp. 120-132. Springer Berlin Heidelberg.
- IA UEA2&UIA, 2006. *Specification of the 3gpp confidentiality and integrity algorithms UEA2 & UIA2*. Document 2: Snow 3g specifications. version: 1.1. etsi.
- Ilyas, M., 2010. *The handbook of ad hoc wireless networks*, Vol. 29. CRC press.
- Koblitz, N., 1987. Elliptic curve cryptosystems. *Mathematics of Computation* 48(177), pp. 203-209.
- Miller, V., 1985. Use of elliptic curves in cryptography. *Advances in Cryptology—CRYPTO'85*, pp. 417-426.
- Rajamani, K., Wentink, M. M., Jones, V. K., 2011. *Wireless display discovery and operation with tdl*. U.S. Patent Application 13/240,852.

- Smart, N. P., 1999. The discrete logarithm problem on elliptic curves of trace one. *Journal of Cryptology*, 12(3), pp. 193–196.
- Viehböck, S., 2011. Brute forcing wi-fi protected setup. http://sviehb.files.wordpress.com/2011/12/viehboeck_wps.pdf.
- Wi-Fi Alliance, 2010. Wi-fi certified wi-fi direct. http://www.wi-fi.org/news_articles.php.
- Wi-Fi Alliance, 2013. <http://www.wi-fi.org/>.

AUTHOR INDEX

| | | | |
|--------------------------|----------|-------------------------|---------------|
| Abdel-Hafez, A. | 184 | Ghorbani, M. | 335 |
| Adán, R. | 57 | Gioia, M. | 359 |
| Ahuja, A. | 14 | Goncalves, N. | 193 |
| Apolloni, B. | 359 | González, M. | 115 |
| Arrue, M. | 65 | Grove, R. | 33 |
| Augello, A. | 344 | Gulla, J. | 278 |
| Baazaoui-Zghal, H. | 123 | HadjKacem, A. | 263 |
| Balík, M. | 107 | Hanrahan, B. | 14 |
| Bardiau, R. | 322 | Hausenblas, M. | 99, 99 |
| Barrera, L. | 82 | Herder, E. | 270 |
| Barrios, A. | 57 | Hernández-Goya, C. | 397 |
| Barros, E. | 328 | Hu, C. | 131 |
| Bartuskova, A. | 143 | Ilarri, S. | 161 |
| Bergamaschi, S. | 172 | Jelínek, I. | 107 |
| Bobed, C. | 41 | Jeon, D. | 313 |
| Borobia, J. | 41 | Jeong, Y. | 313 |
| Boyer, A. | 205 | Ji, Y. | 239, 247 |
| Bsaïes, K. | 205 | Jia, L. | 131 |
| Buey, M. | 161 | Josang, A. | 184 |
| Caballero-Gil, P. | 397 | Juanes, G. | 57 |
| Carrillo-Ramos, A. | 82 | Kalboussi, A. | 263 |
| Chen, W. | 74 | Kanavos, A. | 231, 231 |
| Cheng, X. | 131 | Karagoz, P. | 215 |
| Chiru, C. | 255 | Karnstedt, M. | 99, 99 |
| Cho, Y. | 313 | Kawase, R. | 270 |
| Chou, W. | 74 | Kim, W. | 313 |
| Chunhong, Z. | 247 | Kluth, W. | 149 |
| Ciotec, S. | 255 | Kolas, D. | 33 |
| Decker, S. | 99 | Krejcar, O. | 143 |
| Devlin, K. | 5 | Krempels, K. | 149 |
| Diamanti, K. | 231, 231 | Ksentini, N. | 340 |
| El-Sayed, M. | 295 | Lempesis, E. | 286 |
| Erdur, R. | 278 | Li, L. | 74 |
| Escudero, S. | 161 | Li, Y. | 131, 184 |
| Ferrari, L. | 359 | Liu, X. | 131 |
| Florez-Valencia, L. | 82 | Mahar, K. | 295 |
| Gaggi, O. | 91 | Maier, F. | 371 |
| Galliani, G. | 359 | Makris, C. | 223, 231, 286 |
| García, V. | 27 | Martínez, P. | 115 |
| García, J. | 57 | Mattila, A. | 137 |
| Gargouri, F. | 340 | Mazhoud, O. | 263 |
| Garrido, A. | 41, 161 | Medeiros, C. | 49 |
| Gasparetti, F. | 350 | Medina, L. | 57 |
| Geiger, P. | 383 | Mejia-Molina, N. | 82 |
| Ghezala, H. | 123 | Mena, E. | 41, 161 |

AUTHOR INDEX (CONT.)

| | | | |
|---------------------------|----------|----------------------|----------|
| Micarelli, A. | 350 | Tian, N. | 184 |
| Mikkonen, T. | 137 | Ticha, S. | 205 |
| Mohammadzadeh, H. | 335 | Tmar, M. | 340 |
| Mohasseb, A. | 295 | Tokis, T. | 231 |
| Moreno, L. | 65, 115 | Valencia, X. | 65 |
| Nart, D. | 305 | Videira, A. | 193 |
| Nazemi, A. | 335 | Vilar, B. | 49 |
| Nunes, B. | 270 | Voutilainen, J. | 137 |
| Omheni, | 263 | Wang, Z. | 74 |
| Onal, K. | 215 | Wiegand, N. | 33 |
| Osman, A. | 322 | Wilson, J. | 33 |
| Özgöbek, Ö. | 278 | Xianlei, S. | 247 |
| Panagopoulos, P. | 223 | Xu, Y. | 184 |
| Pavlich-Mariscal, J. | 82 | Yanes, P. | 57 |
| Peiro, A. | 161 | Yang, J. | 239, 247 |
| Pérez-Quiñones, M. | 14 | Yoon, Y. | 313 |
| Pilato, G. | 344 | Zhang, C. | 239 |
| Po, L. | 172 | Zhang, J. | 131 |
| Pryss, R. | 371, 383 | Zhu, Y. | 239 |
| Rakhmawati, N. | 99 | | |
| Rebedea, T. | 255 | | |
| Regazzo, M. | 91 | | |
| Reichert, M. | 371, 383 | | |
| Ribeiro, A. | 328 | | |
| Rivero-García, A. | 397 | | |
| Rodrigues, R. | 27 | | |
| Rojas-Valduciel, H. | 65 | | |
| Roussanaly, A. | 205 | | |
| Samsel, C. | 149 | | |
| Sangiorgi, P. | 344 | | |
| Sansonetti, G. | 350 | | |
| Santanchè, A. | 49 | | |
| Santos-González, I. | 397 | | |
| Schickler, M. | 371, 383 | | |
| Schobel, J. | 371, 383 | | |
| Seguran, M. | 322 | | |
| Seipp, K. | 5 | | |
| Senart, A. | 322 | | |
| Shi, J. | 131 | | |
| Siehndel, P. | 270 | | |
| Silva, C. | 27 | | |
| Sorrentino, S. | 172 | | |
| Souza, R. | 27 | | |
| Systä, K. | 137 | | |
| Tasso, C. | 305 | | |

PROCEEDINGS OF WEBIST 2014 | VOLUME 2

10th International Conference on Web Information Systems and Technologies

ISBN: 978-989-758-024-6 | www.webist.org



Copyright © 2014 SCITEPRESS

Science and Technology Publications

All Rights Reserved

INSTICC is member of:



Logistics Partner:



Proceedings will be submitted for indexation by:



THOMSON REUTERS
CONFERENCE PROCEEDINGS
CITATION INDEX

