

# Seed, an End-user Text Composition Tool for the Semantic Web

Bahaa Eldesouky<sup>1\*</sup>, Menna Bakry<sup>2</sup>, Heiko Maus<sup>1</sup>, and Andreas Dengel<sup>1</sup>

<sup>1</sup> German Research Center for Artificial Intelligence (DFKI),  
Knowledge Management Department,  
Kaiserslautern, Germany,

{bahaa.eldesouky, heiko.maus, andreas.dengel}@dfki.de

<sup>2</sup> German University in Cairo (GUC),  
Cairo, Egypt  
menna.nabil@guc.edu.eg

**Abstract.** Despite developments of Semantic Web-enabling technologies, the gap between non-expert end-users and the Semantic Web still exists. In the field of semantic content authoring, tools for interacting with semantic content remain directed at highly trained individuals. This adds to the challenges of bringing user-generated content into the Semantic Web. In this paper, we present *Seed*, short for Semantic Editor, an extensible knowledge-supported natural language text composition tool for non-experienced end-users. It enables automatic as well as semi-automatic creation of standards based semantically annotated textual content with focus on the task of text composition. We point out the structure of *Seed*, compare it with related work and explain how it excels at utilizing Linked Open Data and state of the art Natural Language Processing to realize user-friendly generation of textual content for the Semantic Web. We also present experimental evaluation results involving a diverse group of 120 participants, which showed that *Seed* helped end-users easily create and interact with semantic content with nearly no prerequisite knowledge.

**Keywords:** semantic web · semantic content authoring · semantic text composition · microdata · lod · nlp · hci

## 1 Introduction

Since the advent of the Semantic Web vision [12], the web has gradually evolved from a structure of interlinked documents to that of interlinked data. This vision drove rapid developments in technologies essential for its realization. Developments in Semantic Web enabling technologies can be seen in the field of modeling and structuring data, where crowd-sourced knowledge repositories such as DB-Pedia [13] and Freebase [14] have grown into huge graphs of entities containing millions of interrelated concepts, which comprise a web of LOD (Linked Open

---

\* Corresponding author.

Data) [8]. In addition to public knowledge repositories, there are private ones, which focus on individual or group knowledge [29] (e.g. corporate knowledge repositories). Also, research on NLP (Natural Language Processing) techniques underwent great developments in both its syntactical and semantic variations [17]. Formats for embedding semantic content in web pages (e.g. RDFa [5], Microformats [4] and Microdata [6]) have also seen growth in their number and adoption rate. However, despite those developments, a gap between non-expert end-users and the Semantic Web still exists. This so-called semantic gap [31] is more evident in the process of creating structured information on the Web, where tools remain directed almost entirely at highly trained individuals [11].

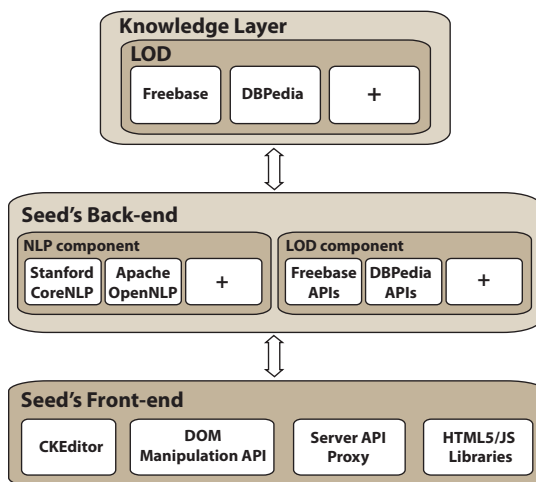


Fig. 1: Architecture of *Seed*

### 1.1 Related Work

Research about technologies that allow for user-friendly consumption and interaction with the existing web of data is gaining traction. SemCards [33] provides an intermediate ontological representational level that allows end-users to create rich semantic networks for their information sphere. OntoAnnotate [32] is an ontology-based annotation environment for web pages based on RDF [25] and RDFschema [15]. RDFauthor [34] bases on making arbitrary XHTML views with integrated RDFa annotations editable. OntosFeeder [23] is a WYSIWYG tool for annotating text for the news/journalism domain. In [7], authors can use Epiphany to get RDFa enhanced versions of their articles that link to Linked Data models.

In [21], the authors surveyed 31 recent primary studies, that dealt with *Semantic Content Authoring (SCA)* of textual content. Special focus was made

on 4 of them, namely OntoWiki [9], [28], SAHA 3 [24], Loomp also known as One Click Annotator [26] and RDFaCE [22]. The authors defined SCA as the tool-supported manual composition process aiming at the creation of documents which have one of two types:

- a) fully semantic (i.e. their original data model uses a semantic knowledge representation formalism).
- b) based on a non-semantic representation enriched with semantic representations during the authoring process. [21, p.2]

Among the variety of SCA tools previously mentioned, the One Click Annotator [19] and RDFaCE have the most similar goals to those of *Seed*. This is why we will closely compare them in a later section.

## 1.2 Contribution

In this paper, we present *Seed*, short for semantic editor, an extensible knowledge-supported Web-based natural language text composition tool, which targets non-experienced end-users. *Seed* aims at bridging the gap between normal users and semantically annotated textual content on the Web. It enables automatic as well as semi-automatic creation of Microdata-annotated [6] HTML-based textual content without any domain knowledge requirements regarding the underlying technology or annotation formats. We point out the structure of *Seed*, explain how it builds upon developments in the fields of NLP, LOD and other Semantic Web technologies to provide a user-friendly way of creating and interacting with knowledge on the Web. We contrast *Seed* with comparable works and show what distinguishes it:

As discussed and later demonstrated through experimental evaluation in this paper, we show that:

- *Seed*'s focus on non-expert end-users makes semantics completely transparent to authors by focusing on the process of text composition, the actual interest of end-users, rather than semantic annotation or the underlying semantic analysis.
- It realizes more aspects of end-user SCA systems mentioned in [21] such as:
  - Real-time annotation during composition, which encourages users to review and interact with annotations making them more reliable. In that regard, comparable systems, are better described as a posteriori annotation tools.
  - Inline annotations behave like normal text reacting to inserting, deleting or updating characters all while preserving correct clean semantic markup.
  - Going beyond annotation to enable interaction and exploration of knowledge
- It is rigorously evaluated in terms of scale of the evaluation and evaluated aspects diversity, a much needed practice in Semantic Web research. Our experimental user study involved 120 participants from various backgrounds,

age-groups and nationalities, we evaluated the usability of *Seed*, the quality of content it produces, and the subjective opinion of participants about the value of using *Seed* to explore, modify, and create semantic content.

This paper proceeds as follows: Section 2 explains the architecture and implementation of *Seed* showing how it builds on open Web technologies and standards. Section 3 assesses *Seed* in comparison with two related works and points out what distinguishes it as a SCA tool. In section 4, we discuss in detail the setup and results of an experimental evaluation study. Finally, we wrap up and present examples of future work in the conclusion.

## 2 Seed Architecture and Implementation

As shown in Fig. 1, *Seed* consists of 3 loosely-coupled main components: a) Knowledge layer, b) Back-end and c) Web front-end. Mutual communication between these components uses standard Web APIs (e.g. REST -based [18] Web services) to promote interoperability.

### 2.1 Knowledge layer

This is a logical component of *Seed*, which represents the collective body of structured information available on the Semantic Web. Possible sources of knowledge integrable in this layer include not only public LOD sources such as DBPedia, but also any ontology-based knowledge repository. The current implementation of *Seed* integrates two LOD sources in its knowledge layer, namely DBPedia and Freebase. Other knowledge sources can be integrated as shown in Fig. 1.

### 2.2 Back-end

The second logical component of *Seed* is subdivided into two sub-components:

**NLP Component** This sub-component utilizes state-of-the-art NLP toolkits to perform tasks such as part of speech tagging (POS), named entity recognition (NER), coreference resolution, ... etc. The implementation is carried out in a modular way that eases integrating or swapping various NLP toolkits as implied in Fig. 1. The current implementation of *Seed* specifically builds upon Stanford CoreNLP [27] and Apache OpenNLP [1] to provide a server API capable of real-time analysis of the text being authored. It also extracts named entity candidates, which are then processed to discover knowledge. This component currently supports English and German.

**LOD Component** Together with the NLP component, this component communicates in real-time with LOD sources to extract information about potential entities extracted from the text. It is responsible for performing entity disambiguation and providing contextual information about discovered entities. The

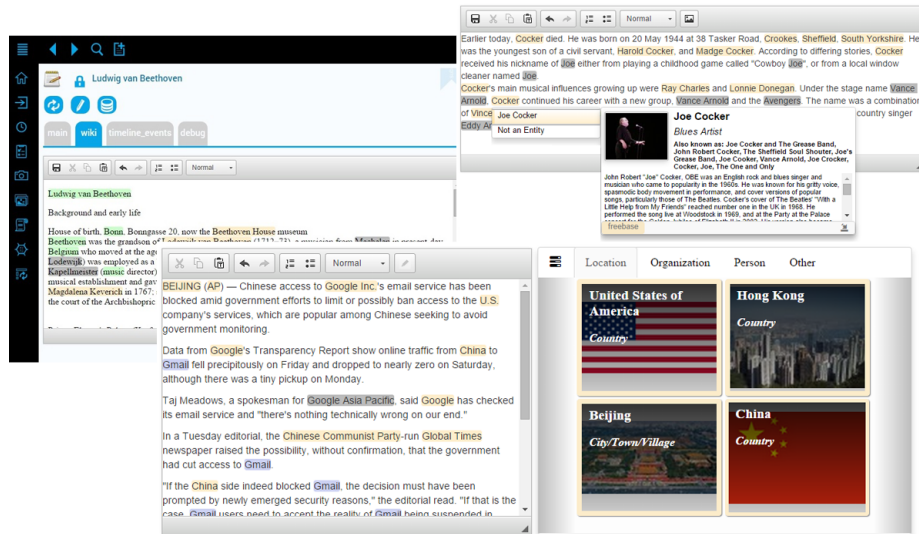


Fig. 2: Screenshots of multiple configurations of *Seed*'s front-end

LOD component does not enforce a specific vocabulary or domain on the front-end. This has the following benefits for end-user oriented use-cases:

- freeing the creator of the user interface (UI) in which *Seed* is to be embedded, from being restricted by the back-end's choice and
- elimination of the mental overload on the end-user incurred in understanding the vocabulary and learning to use it.

However, it is also possible to extend the front-end to enforce a specific vocabulary if the application domain or the use-case at hand dictates it.

### 2.3 Web Front-end

The current prototype of *Seed*'s front-end is meant to run in the browser (see Fig. 2 for various configurations of *Seed* in the browser). It is implemented as a pluggable component suitable for any Web-based UI. Therefore, it is written completely in HTML5 and JavaScript (JS). Nonetheless, it is also possible to embed it in non-Web GUIs. The only prerequisite is the availability of an HTML capable UI element. This flexibility in integrating *Seed* makes it highly portable and facilitates reaching end-users dealing with different UI types or different types of devices. The front-end component consists of the following logical sub-components.

**CKEditor** At the core of the front-end, *Seed* builds upon CKEditor [2], the open source WYSIWYG HTML editor. We have extended CKEditor with the following components:



Fig. 3: Main parts of *Seed*'s front-end: 1- Dropdown menu for viewing, selecting or rejecting an annotation, 2- Suggested as well as confirmed inline annotations, 3- Entity side pane for viewing more information about entities in the text, 4- Controls for Faceted viewing/browsing, 5- Text composition area

**DOM Manipulation API** In order to implement inline editing of HTML content including semantic markup in a reliable usable way, we have built upon JavaScript (JS) native HTML Document Object Model (DOM) manipulation constructs, jQuery [3] as well as CKEditor's own JS APIs to implement a basic API for monitoring and interacting with the HTML DOM for text editing purposes.

**Server API Proxy** This is JS code that handles communication with the server in near real-time. It consumes standard RESTful APIs provided by the NLP and LOD components of *Seed*'s back-end to update the semantic representation of the text as it changes.

**Semantic annotator** The semantic annotator is a JS/HTML extension code responsible for:

- building upon the DOM manipulation API of *Seed* to add, remove or update Microdata annotations during editing. Annotations are applied in the form of HTML Microdata markup,
- maintaining a client side representation of the knowledge in the text in the form of entities and their metadata,
- binding between entities and their arbitrary UI manifestations (labels, highlights, information panes, images, ... etc).

**HTML5/JS Libraries** For the creation of the various elements of the front-end, we have used the following main third-party JS/HTML5/CSS libraries: (jQuery, jQuery Mobile, Mutation Summary and Bootstrap)

As shown in Fig. 3, various parts of the *Seed*'s UI allow authors to interact with the textual and knowledge content of the text being composed.

### 3 Comparative Assessment

Despite obviously increasing research interest in the field of SCA, our literature survey of related work pointed to only a handful of approaches to SCA, which focus on semantic text composition of natural language text. Fewer are works, which target end-users in contrast to Semantic Web computer professionals.

In this section, we will assess *Seed* in comparison with two other SCA tools (One Click Annotator and RDFaCE) due to their conceptual similarity. Both tools also follow a bottom up semantic authoring approach like *Seed*. They target, at least in part, end-users. Our assessment will start by a tabular comparison inspired by [21]. We will augment the quality attributes for assessing SCA systems listed by the authors with additional metrics we suggest to form a basis for the comparison. According to [21], those quality attributes adopt the point of view of SCA users (i.e. end-users in our case). For each quality attribute, concrete UI features should realize the respective quality attribute [21, p.7].

After the tabular comparison, we will discuss selected comparative aspects in detail to point out the significance of *Seed*.

Table 1: SCA quality attributes assessment of *Seed*. Table layout adapted from [21, p.12]

|                                    | One Click Annotator                                    | RDFaCE                      | <b>Our system (Seed)</b>                                       |
|------------------------------------|--|-----------------------------|--|
| Online demo                        | N/A  | rdface.aksw.org/            | tiny.cc/seed-demo  |
| Experimental evaluation            | 12 non-expert participants, paper prototype user study | 16 experienced participants | <b>120 non-expert &amp; expert subjects, online experiment</b> |
| Usability                          |  |                             |  |
| Single entry point                 | yes  | yes                         | yes  |
| Faceted browsing                   | no   | no                          | <b>yes</b>   |
| Faceted viewing                    | yes  | yes                         | yes  |
| Inline editing                     | no   | yes                         | yes  |
| View Editing                       | no   | no                          | <b>yes</b>   |
| Native annotations                 | no   | no                          | <b>yes</b>   |
| Automation                         |  |                             |  |
| Automatic annotation               | no   | external NLP APIs           | own server API   |
| Generalizability                   |  |                             |  |
| Multiple ontologies support        | yes  | yes                         | yes  |
| Ontology modification support      | no   | no                          | no   |
| Heterogeneous content formats      | no   | no                          | no (only standard HTML5)                                       |
| Collaboration                      |  |                             |  |
| Access control                     | no   | no                          | <b>yes</b>   |
| Standard formats support           | RDF & RDFa   | RDFa & Microdata            | Microdata  |
| UIs for social collaboration       | no   | no                          | (depends on Web UI)  |
| Customizability                    |  |                             |  |
| Living UIs                         | no   | no                          | no   |
| Providing different semantic views | no   | yes                         | yes  |
| Evolvability                       |  |                             |  |

|                                   |                          |                                     |   |
|-----------------------------------|--------------------------|-------------------------------------|---|
| Resource consistency              | no                       | no                                  | yes   |
| Document & annotation consistency | yes                      | yes                                 | yes   |
| Versioning                        | no                       | no                                  | yes   |
| Proactivity                       |                          |                                     |   |
| Resource suggestion               | yes                      | yes                                 | yes   |
| Real-time semantic tagging        | no                       | no                                  | yes   |
| Concept reuse                     | yes                      | no                                  | yes   |
| Real-time validation              | no                       | no                                  | yes   |
| Portability                       |                          |                                     |   |
| Cross-browser compatibility       | yes                      | yes                                 | yes   |
| Mobile UI support                 | no                       | no                                  | no  |
| Accessibility                     |                          |                                     |   |
| Accessible UIs                    | no                       | no                                  | no  |
| Interoperability                  |                          |                                     |   |
| Standard formats                  | RDF & RDFa               | RDFa & Microdata                    | Microdata   |
| Semantic Syndication              | no                       | no                                  | no  |
| Scalability                       |                          |                                     |   |
| Caching support                   | no                       | no                                  | no  |
| Storage strategy                  | server-side triple store | on the fly client-side triple store | <b>triple store or live LOD + client-side in-memory storage</b> |

A review of the condensed comparison in Table 1 reveals many advantages of *Seed* over One Click Annotator and RDFaCE. For the sake of brevity, we will elaborate on some of those advantages and suffice to the table entries for the rest.

**No prerequisite knowledge:** *Seed* requires no prerequisite technical knowledge about formats of embedding semantic content, underlying knowledge representation models, vocabularies or even the basic terminology of the Semantic Web. For example, annotating text with information about its semantics, modifying existing annotations and exploring knowledge about entities beyond the textual content being authored take place through familiar user interaction scenarios. In contrast, other similar works vary from requiring knowledge of triple representations to learning specific vocabularies.

**Real-time annotation:** One Click Annotator and RDFaCE require text authors to explicitly request the annotation of content. This reduces the productivity of the text composition task and increases the mental load on authors, which in turn discourages end-users from reviewing and possibly correcting annotations of the content. *Seed*, on the other hand, continuously analyzes authored text and proactively annotates it with semantic information. The reduced effort required from authors, is expected to encourage them to review and interact with annotations, thus producing more reliable semantic annotations.

**Native annotations:** In *Seed*, annotations behave like normal text. They react to inserting, deleting or updating characters intuitively and consistently, thus producing correct semantic markup. In comparable systems like RDFaCE, once annotations are created, attempts to modify them break the semantic markup.

**Focus on knowledge:** In addition to generating semantic content, *Seed* focuses on enabling users to consume the underlying knowledge in a high level



fashion and through different views. As shown in Fig. 2 and Fig. 3, it provides multiple views to the underlying semantics on annotations. By means of a faceted view, authors can have a high level view of the entities mentioned in the text and are able to explore information about the entities derived from LOD sources.

**Rigorous evaluation:** To our knowledge, no other comparable system has been as rigorously evaluated as *Seed*, neither in terms of the scale of the evaluation, the target test-subject audience or the diversity of evaluated aspects. RDFaCE for example, has been evaluated by 16 participants from a computer science background participating in a LOD workshop [22].

## 4 Experimental evaluation

A comparative experimental evaluation with other works such as RDFaCE and One Click Annotator, which involved 120 participants was not feasible in the scope of our study. Reasons include the lack of publicly accessible functioning prototypes/demos of other works. Besides, the scale of the experiment and the practical time limits for an online evaluation made it impossible for us to evaluate other works using the same procedure without substantially shrinking the population. So, we focused instead on evaluating *Seed* while providing enough information for reproducing the evaluation by others <sup>3</sup>.

### 4.1 Goals

As previously mentioned, an important yet missing aspect of research on SCA is user studies of sizable scale involving ordinary non-expert users. Most of the studies in the field propose conceptual ideas, which are seldom put to reasonable evaluation. Therefore, we have set out to target a large group of Web-users with the following goals in mind:

1. Show that *Seed* is a highly usable and easy-to-learn semantic text composition tool, which hides the complexity of the underlying technology, thus enabling Semantic Web end-users to focus on the process of textual content generation.
2. Enable end-users with no prerequisite knowledge to produce standards based semantically annotated textual content.
3. Proactively help end-users to explore, and interact with knowledge from the Semantic Web (LOD in our case) while composing textual content.

### 4.2 Design

The evaluation was designed as a within-subjects repeated measures experiment. All participants were exposed to the same conditions. The independent variables of the study were:

1. The number of text passages
2. The length of each passage
3. The number of entities in each passage

<sup>3</sup> Evaluation data available at <http://tiny.cc/seed-iswc2016-data>

### 4.3 Procedure

We have set up an evaluation website at <http://tiny.cc/seed-demo> and prepared the experiment, which consisted of the following stages:

- User registration, where participants were asked to provide information about themselves for demographic profiling and validation purposes.
- Once registered, participants watched a 3 min. video<sup>1</sup> that explained the concept of *Seed* in a non-technical way. We refrained from detailed descriptions of technical aspects of the system in order to properly measure its learnability by non-experts.
- Participants were then asked to review and annotate 3 text passages using *Seed*. Every participant started with a pre-loaded text. The user then reviewed automatic annotations by *Seed* as well as annotation suggestions that (s)he could confirm, modify, reject or augment.
- Afterwards, participants were asked to type in a predefined text passage into *Seed*, which gets annotated in real time and reviewed during writing. Then, participants are asked questions to test their understanding of the text and validate their attention to the experiment. Answers helped us later pre-process data and eliminate non-serious participants.
- Finally, participants were asked to fill in a standardized usability questionnaire to assess participants’ satisfaction with the perceived usability of the system, then answer additional questions about *Seed*.

### 4.4 Participants

The evaluation received 256 registrations, of which 120 completed the experiment. Table 2 shows demographic information about participants.

Table 2: Demographics of the participants population

| Characteristics | Percentages   |
|-----------------|---|
| Age             | 15-25 years (47%), 25-35 years (42%), 35-45 years (9%), 55-65 years (2%)  |
| Gender          | Males (61%), Females (39%)  |
| Profession      | Undergraduate students (31%), graduate students (23%), computer professionals (20%), non computer professionals (19%), researchers (7%) |
| Nationality     | Egypt (59%), Germany (14%), India (8%), Jordan (3%), Pakistan (3%), Palestine (2%), Others (11%)  |

### 4.5 Measures

For the assessment of semantic annotations, we calculated precision, recall and F-1 scores of annotations done by participants for all texts. For assessing learnability, we measured the time required for reviewing and annotating texts in the repeated measures part of the experiment. For assessing the perceived usability of the system, we measured the System Usability Scale score (SUS) for *Seed*

<sup>1</sup> Seed, the Semantic Editor - <http://tiny.cc/seed-video>

#### 4.6 Semantic Annotations Assessment

For the choice of texts to be authored and annotated in the experiment, we provided all participants with a set of text passages to achieve as much consistency as possible in regard to the length of text, the number of entities mentioned, their types and the subject domain. To compensate for the small size of dataset we targeted a large participants population. The outcome of the experiment was then assessed against a ground truth version of the set of texts.

The texts used in the experiment were produced as follows. We selected 3 representative text passages from different subject domains (news articles, wiki articles, and blog posts) to control the subject domain familiarity variable. The fourth passage was arbitrarily selected to be from the wiki articles domain.

To create a ground truth for assessing annotations in the text passages, 3 different human annotators separately annotated named entities of type person, location or organization by hand. We restricted types of annotations in the ground truth to the three mentioned types to parallel the most widely used 3-class model in state of the art NLP tools. Only annotations agreed upon by 2 or more annotators were added.

In order to evaluate *Seed*'s ability to produce correct annotations during text composition, we calculated Precision, Recall and F-1 scores for annotations in all passages submitted by participants as shown in Table 3.

Table 3: Annotation performance measures assessment

|           | Avg. Recall | Avg. Precision | Avg. F1 |
|-----------|-------------|----------------|---------|
| Passage 1 | 0.80        | 0.79           | 0.79    |
| Passage 2 | 0.89        | 0.97           | 0.93    |
| Passage 3 | 0.76        | 0.88           | 0.80    |
| Passage 4 | 0.89        | 0.93           | 0.91    |

For the calculation of the performance measure values, we considered an entity annotation correct if it was:

- correctly recognized (i.e. there is an entity and its token delimiters were identified by the author)
- disambiguated and correctly mapped to a LOD entity from DBpedia or Freebase

The values in Table 3 show that *Seed* helped automatically annotate the majority of the entities in the text already at the text authoring stage.

The following interesting observations resulted from the performance measures assessment for all texts:

On average 38.2% percent of participants annotated more entities in total than existed in the ground truth. This is a valuable remark because it shows that *Seed*'s support for annotation during text composition goes beyond the limitations of state-of-the-art NLP models. On average 13.5% percent of participants submitted more correct annotations than the total number of annotations in the ground truth. This shows that the annotations submitted by participants are not only more but also correct.

#### 4.7 Usability Evaluation

**System Usability Scale Score** At the end of the experiment, we prompted users to fill in a questionnaire which consisted of a standard SUS form in addition to two questions we added. As defined by [16], scores of individual items in a SUS are not meaningful on their own. So, we calculated the overall SUS score for *Seed* across the population of participants, which resulted in an overall SUS score with mean: 73.56, median: 75, standard deviation of 13.71. According to [16], this means *Seed* has above average usability. In order to assess the statistical significance of the SUS results, we performed a one sample Z-test on the SUS scores of the participants.

Following Sauro’s notion in [30], we defined our hypotheses as follows:

- Null Hypothesis,  $H_0$ : It’s predicated that Seed’s SUS score is at most around average ( $\mu \leq 70$ ).
- Alternate hypothesis,  $H_a$ : Seed’s SUS score above average ( $\mu > 70$ ).

The results of the Z-test showed that SUS scores for Seed in our experiment ( $\mu = 73.96, \sigma = 13.94$ ) are significantly higher than the predicated SUS score of 70 ( $z = 2.71, p = 0.0034$ ). According to [10], we can confidently say that Seed’s SUS score is between good and excellent.

**Interactive Inline Annotations** In order to assess the effect of interactive inline annotations in authored text in *Seed* on its overall usability, we asked participants the following two alternating tone questions with answers varying on a 5-degree scale (0 to 4), from “strongly disagree” to “strongly agree” respectively:

- The annotated entities helped me to understand the written content
- Entities annotated in the text distracted me from reading the content

A clear majority of users found the annotations not distracting. The answers to the negatively formulated question had a median= 1 and a mean= 1.42. They also found them helpful in understanding the content of the text they were annotating. Answers to the positively formulated question had a median= 3 and a mean= 2.85.

**Real-time Annotation** In order to assess the value of real-time annotation, we asked participants the following question:

“Which of the following options would you prefer more?”

- a) Annotating entities as you write them
- b) Annotating once after you finish typing the text
- c) No preference

According to responses, 55.5555%  $\simeq$  55.6% of the participants chose a), 38.8888%  $\simeq$  38.9% chose b) while 5.5555%  $\simeq$  5.6% expressed no preference. It is worth mentioning that many of the users who chose b) justified their choice by the relative simplicity of the topic of the 3 passages or by the irrelevance to a personal context of theirs.

**Faceted Viewing and Knowledge Discovery** These important features of *Seed*'s UI aim at enabling end-users to easily explore and consume knowledge about content of the text being authored. In order to evaluate these features, we asked users questions whose answers are not contained in a passage, but are available through the entity summary side pane of *Seed* as well as through the interactive annotation information pane. The results of users answers were as follows:

- For the questions, whose answers required looking into the information in the entity side pane or in the interactive annotation info pane, 94.9% of the participants managed to find the correct answer.
- For the question, whose answer is most easily accessible by faceted browsing, 51.5% of the participants managed to find the answer.

To check whether participants had looked up the answer elsewhere, we asked them how they found it. For those who correctly answered at least one question, 93.9% did so using *Seed*'s features. This showed that *Seed* successfully helped participants discover knowledge about the content. The results hint at the need for further inspection of the design of the faceted browsing feature.

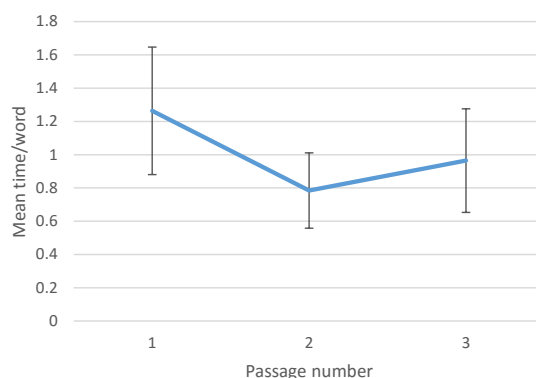


Fig. 4: Mean time per word required to annotate text passages. An overall decreasing trend is seen, which signals the speed of learning of users

**Learnability** To assess how fast participants learned to use *Seed*, we carried out the following:

- We measured the time required for annotating each of the first 3 text passages for all of participants.
- Outliers were eliminated using a two-sided Iglewicz and Hoaglin's robust test for multiple outliers [20].

- To account for varying length of the texts, we calculated the time per word in each passage.
- In order to check for statistically significant differences in mean times per word required for annotating texts, a repeated measures ANOVA with a Greenhouse-Geisser correction determined that mean time per word differed statistically significantly between passages ( $F(1.89, 177.696) = 17.09, P < .0005$ ). Post hoc tests using the Bonferroni correction revealed that time/word decreased from passage 1 to passage 2 ( $1.26 \pm 0.77s$  vs  $0.79 \pm 0.45s$ , respectively), which was statistically significant ( $p < .0005$ ). Also, time/word decreased from passage 1 to passage 3 ( $1.26 \pm 0.77s$  vs  $0.96 \pm 0.62s$ , respectively), which was also statistically significant ( $p = 0.005$ ). However, time/word slightly increased from passage 2 to passage 3 ( $0.79 \pm 0.45s$  vs  $0.96 \pm 0.62s$ , respectively), which was not statistically significant ( $p = 0.055$ ). Therefore, we can conclude that an overall decreasing trend exists from passage 1 on one hand and passage 2 or 3 on the other hand.

To explain the slight increase in passage 3 time, we further inspected annotation data and qualitative feedback collected in the experiment. Deteriorating performance measures for passage 3 combined with re-occurring comments regarding passage 3 about the inability to annotate overlapping entities such as “Old Town Hall” and “New Town Hall” in a sentence containing the text “Old and New Town Hall” provide an explanation for the apparent increase. It also highlights a technical limitation in dealing with overlapping entities in HTML based semantic annotations. Attempts to annotate pairs of overlapping entities is not easily doable in HTML markup due to its hierarchical nature.

## 5 Conclusion

In previous sections, we highlighted the importance of bridging the gap between end-users and the Semantic Web. We presented *Seed*, a user-friendly semantic text composition tool, which brings technical non-experts closer to the Semantic Web. It allows them to benefit from, interact with, and create semantic content in the form of semantically annotated HTML-based text. We showed how it realizes real-time annotation during authoring, thus encouraging end-users to review, possibly add annotations as they write. Using rigorous experimental evaluation involving a sizable, diverse population of 120 participants, we assessed our hypotheses about *Seed*. Results showed that it enabled users to produce semantically annotated textual content in a reliable way. By means of a standard SUS evaluation, *Seed* proved highly usable, not only in annotating content, but also in exploring knowledge about it.

The outcome of this paper gives insight into future work research questions. The loosely-coupled architecture of *Seed* combined with the fact that it supports German as well as English, encourages us to explore its use for multilingual content. *Seed*'s ability to integrate with public knowledge sources motivates us to explore its use in application scenarios where personal rather than public knowledge is more relevant. Also, exploring richer semantic representations embedded

in the text (e.g. relations between entities) is an interesting possibility. This in turn will further contribute to bridging the gap between end-users and the Semantic Web.

## 6 Acknowledgement

The work presented in this paper was partially supported by the supSpaces project funded by the German Federal Ministry of Education and Research (BMBF) under grant no. 01IS15013B.

## References

1. Apache OpenNLP. <http://opennlp.apache.org/index.html>, “Online, accessed 18 July 2016”
2. CKEditor. <http://ckeditor.com>, “Online, accessed 18 July 2016”
3. jQuery. <http://jquery.com/>, “Online, accessed 18 July 2016”
4. Microformats. <http://microformats.org/>, “Online, accessed 18 July 2016”
5. RDFa Core 1.1 - Second Edition. <http://www.w3.org/TR/rdfa-syntax/> (22-August-2013), “Online, accessed 18 July 2016”
6. HTML Microdata. <http://www.w3.org/TR/microdata/> (29-October-2013), “Online, accessed 18 July 2016”
7. Adrian, B., Hees, J., Herman, I., Sintek, M., Dengel, A.: Epiphany: Adaptable RDFa Generation Linking the Web of Documents to the Web of Data. In: Knowledge Engineering and Management by the Masses, pp. 178–192. Springer (2010)
8. Auer, S., Bryl, V., Tramp, S.: Linked Open Data—Creating Knowledge Out of Interlinked Data: Results of the LOD2 Project, vol. 8661. Springer (2014)
9. Auer, S., Dietzold, S., Riechert, T.: OntoWiki – A Tool for Social, Semantic Collaboration. In: The Semantic Web-ISWC 2006, pp. 736–749. Springer (2006)
10. Bangor, A., Kortum, P., Miller, J.: Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of usability studies* 4(3), 114–123 (2009)
11. Benson, E., Karger, D.R.: End-users Publishing Structured Information on the Web: An Observational Study of What, Why, and How. In: Proceedings of the 32nd annual ACM conference on Human factors in computing systems. pp. 1265–1274. ACM (2014)
12. Berners-Lee, T., Hendler, J., Lassila, O., et al.: The Semantic Web. *Scientific american* 284(5), 28–37 (2001)
13. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - A Crystallization Point for the Web of Data. *Web Semantics: science, services and agents on the world wide web* 7(3), 154–165 (2009)
14. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In: Proceedings of the 2008 ACM SIGMOD international conference on Management of data. pp. 1247–1250. ACM (2008)
15. Brickley, D., Guha, R.V.: Resource Description Framework (RDF) Schema Specification 1.0: W3C Candidate Recommendation 27 March 2000 (2000)
16. Brooke, J.: SUS - A Quick and Dirty Usability Scale. In: Jordan, P., Thomas, B., Weerdmeester, B.A., McClelland, I. (eds.) *Usability Evaluation in Industry*, vol. 189. Taylor and Francis, London (1996)

17. Cambria, E., White, B.: Jumping NLP curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine* 9(2), 48–57 (2014)
18. Fielding, R.T.: Architectural Styles and the Design of Network-based Software Architectures. Ph.D. thesis, University of California, Irvine (2000)
19. Heese, R., Luczak-Rösch, M., Oldakowski, R., Streibel, O., Paschke, A.: One click annotation. In: *CEUR Workshop Proceedings*. vol. 514 (2009)
20. Iglewicz, B., Hoaglin, D.C.: How to detect and handle outliers, vol. 16. *Asq Press* (1993)
21. Khalili, A., Auer, S.: User interfaces for semantic authoring of textual content: A systematic literature review. *Web Semantics: Science, Services and Agents on the World Wide Web* 22, 1–18 (Oct 2013)
22. Khalili, A., Auer, S., Hladky, D.: The RDFa Content Editor – From WYSIWYG to WYSIWYM. In: *Computer Software and Applications Conference (COMPSAC), 2012 IEEE 36th Annual*. pp. 531–540. *IEEE* (2012)
23. Klebeck, A., Hellmann, S., Ehrlich, C., Auer, S.: OntosFeeder – A Versatile Semantic Context Provider for Web Content Authoring. In: *The Semantic Web: Research and Applications*, pp. 456–460. *Springer* (2011)
24. Kurki, J., Hyvönen, E.: Collaborative Metadata Editor Integrated with Ontology Services and Faceted Portals. *ORES-2010 Ontology Repositories and Editors for the Semantic Web* p. 7 (2010)
25. Lassila, O., Swick, R.R., et al.: *Resource Description Framework (RDF) Model and Syntax Specification* (1998)
26. Luczak-Rösch, M., Heese, R.: Linked Data Authoring for Non-Experts. In: *LDOW* (2009)
27. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. pp. 55–60 (2014)
28. Salas, P.E.R., Martin, M., Da Mota, F.M., Auer, S., Breitman, K.K., Casanova, M.A.: OLAP2DataCube: An Ontowiki Plug-In for Statistical Data Publishing. *TOPI2012* pp. 79–83 (2012)
29. Sauermann, L., van Elst, L., Dengel, A.: PIMO – A Framework for Representing Personal Information Models. In: Pellegrini, T., Schaffert, S. (eds.) *I-SEMANTICS Conference 5-7 September 2007, Graz, Austria*. pp. 270–277. *J.UCS, Know-Center, Austria* (2007)
30. Sauro, J.: *A practical guide to the system usability scale: Background, benchmarks & best practices*. *Measuring Usability LLC* (2011)
31. Siorpaes, K., Simperl, E.: Human Intelligence in the Process of Semantic Content Creation. *World Wide Web* 13(1-2), 33–59 (2010)
32. Staab, S., Maedche, A., Handschuh, S.: *Creating Metadata for the Semantic Web – An Annotation Environment and the Human Factor*. *Institute AIFB* (2000)
33. Thórisson, K.R., Spivack, N., Wissner, J.M.: SemCards: A New Representation for Realizing the Semantic Web. In: *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, pp. 425–436. *Springer* (2009)
34. Tramp, S., Heino, N., Auer, S., Frischmuth, P.: RDFauthor: Employing RDFa for Collaborative Knowledge Engineering. In: *Knowledge Engineering and Management by the Masses*, pp. 90–104. *Springer* (2010)