# A Probabilistic Model for Time-Aware Entity Recommendation

Lei Zhang, Achim Rettinger, and Ji Zhang

Institute AIFB, Karlsruhe Institute of Technology (KIT), Germany
l.zhang@kit.edu

**Abstract.** In recent years, there has been an increasing effort to develop techniques for related entity recommendation, where the task is to retrieve a ranked list of related entities given a keyword query. Another trend in the area of information retrieval (IR) is to take temporal aspects of a given query into account when assessing the relevance of documents. However, while this has become an established functionality in document search engines, the significance of time has not yet been recognized for entity recommendation. In this paper, we address this gap by introducing the task of *time-aware entity recommendation*. We propose the first probabilistic model that takes time-awareness into consideration for entity recommendation by leveraging heterogeneous knowledge of entities extracted from different data sources publicly available on the Web. We extensively evaluate the proposed approach and our experimental results show considerable improvements compared to time-agnostic entity recommendation approaches.

## 1 Introduction

The ever-increasing quantities of entities in large knowledge bases on the Web, such as Wikipedia, DBpedia and YAGO, pose new challenges but at the same time open up new opportunities of information access on the Web. In recent years, many research activities involving entities have emerged and increasing attention has been devoted to technologies aimed at identifying entities related to a user's information need. *Entity search* has been defined as finding an entity in the knowledge base that is explicitly named in a keyword query [1]. A variant of entity search is *related entity recommendation*, where the goal is to rank relationships between a query entity and other entities in a knowledge base [2,3]. In the context of Web search, *entity recommendation* has been defined as finding the entities related to the entity appearing in a Web search query [4].

On the other hand, temporal dynamics and their impact on information retrieval (IR) have drawn increasing attention in the last decade. In particular, the study of document relevance by taking into account the temporal aspects of a given query is addressed within *temporal IR* [5]. To support a temporal search, a basic solution is to extend keyword search with the creation or publication date of documents, such that search results are restricted to documents from a particular time period given by a time constraint [6,7]. This feature is already

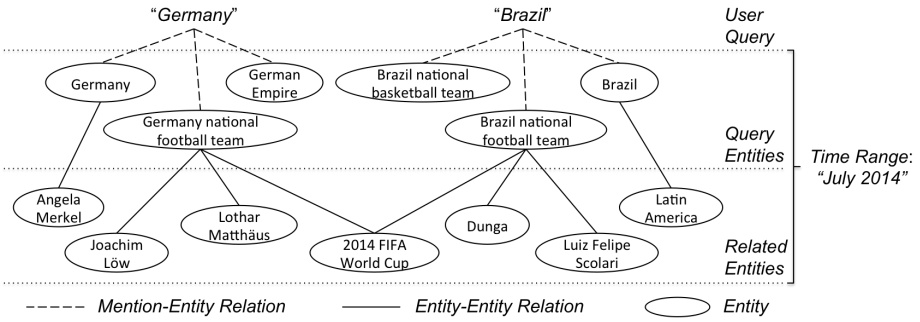


Fig. 1: Examples of the query and related entities for the user query "*Germany Brazil*" and the given time range "*July 2014*".

available in every major search engine, e.g., Google also allows users to search Web documents using a keyword query and a customized time range. For the effectiveness of temporal IR, the time dimension has been incorporated into retrieval and ranking models, also called *time-aware retrieval and ranking*. More precisely, documents are ranked according to both textual and temporal similarity w.r.t. the given temporal information needs [5].

Inspired by temporal IR, we believe that the time dimension could also have a strong influence on entity recommendation. Existing entity recommendation systems aim to link the initial user query to its related entities in the knowledge base and provide a ranking of them. Typically, this has been done by exploiting the relationships between entities in the knowledge base [2,3,4]. However, the (temporal) entity importance and relatedness is often significantly impacted by real-world events of interest to users. For example, a sports tournament could drive searches towards the teams and players that participate in the tournament and the acquisition of a company by another company could establish a new relationship between them and thus affect their relatedness. Some efforts have already been devoted to improve the quality of recommendations in particular with respect to data freshness. For example, Sundog [8] uses a stream processing framework for ingesting large quantities of Web search log data at high rates such that it can compute feature values and entity rankings in much less time compared to previous systems, such as Spark [4], and thus can use more recently collected data for the ranking process. However, the time-awareness, which should be a crucial factor in entity recommendation, has still not been addressed.

Let us suppose users issue the keyword query "*Germany Brazil*" (see Fig. 1). Then they are likely looking for related geographic or political entities. However, when additionally specifying the time range "*July 2014*", their interest is more likely related to the German and Brazilian national football teams during the 2014 FIFA World Cup. Obviously, once time information is available, the goal for a related entity search approach should be to improve entity recommendation such that the ranking of related entities depends not only on entity information in the knowledge base but also on the real-world events taking place in a specific time period. Therefore, it is essential to make *time-awareness* a top priority in entity recommendation when a customized time range is given.

In this paper, we introduce the problem of *time-aware entity recommendation* (TER), which allows users to restrict their interests of entities to a customized time range. In general, the goal of TER is to (1) *disambiguate the query entities* mentioned in the user query and (2) *find the related entities* to the query entities as well as (3) *rank all these query entities and related entities according to time* in order to match information needs of users, where the time dimension plays an important role. As shown in Fig. 1, the keywords "*Germany*" and "*Brazil*" result in different potential query entities. Since Germany_national_football_team and Brazil_national_football_team are of particular interest during the given time range "*July 2014*", they should more likely be the intended query entities. For each query entity, its related entities will be found through the relations between entities, which can also be influenced by the time dimension. For example, the query entity Brazil_national_football_team results in the related entities Dunga, the current coach of Brazilian national football team, and Luiz_Felipe_Scolari, the coach during 2014 FIFA World Cup. By taking into account the time dimension, Luiz_Felipe_Scolari should be preferred over Dunga since the user requests information from July 2014.

To achieve this, we propose a probabilistic model by decomposing the TER task into several distributions, which reflect heterogeneous entity knowledge including *popularity*, *temporality*, *relatedness*, *mention* and *context*. The parameters of these distributions are then estimated using different real-world data sources, namely Wikipedia[1], Wikilinks[2], Wikipedia page view statistics[3] and a multilingual real-time stream of annotated Web documents. Please note that the data sources used by existing systems are mostly not publicly accessible. Particularly the major Web search engines keep their own usage data, like query terms and search sessions as well as user click logs and entity pane logs, secret, since they are crucial to optimizing their own entity recommendation systems, like the ones of Yahoo! [4,8] and Microsoft [9,10]. In contrast, our approach does not rely on datasets taken from commercial Web search engines, but only resorts to data sources publicly available on the Web.

The main contributions of this paper are: (1) We introduce a formal definition of the TER problem (2) and propose a statistically sound *probabilistic model* that incorporates *heterogeneous entity knowledge* including the *temporal context*. (3) We show how all parameters of our model can be effectively estimated solely based on data sources *publicly available* on the Web. (4) Due to the lack of benchmark datasets for the TER challenge, we have created *new datasets* to enable empirical evaluations and (5) the results show that our approach improves the performance considerably compared to time-agnostic approaches.

The rest of the paper is organized as follows. We present the overall approach, especially the probabilistic model in Sec. 2. Then, we describe the estimation of model parameters in Sec. 3. The experimental results are discussed in Sec. 4. Finally, we survey the related work in Sec. 5 and conclude in Sec. 6.

[1] `https://dumps.wikimedia.org/`
[2] `http://www.iesl.cs.umass.edu/data/wiki-links/`
[3] `https://dumps.wikimedia.org/other/pagecounts-raw/`

## 2 Approach

We first formally define the *time-aware entity recommendation* (TER) task and then describe the probabilistic model of our approach.

**Definition 1 (Time-Aware Entity Recommendation).** *Given a knowledge base with a set of entities $E = \{e_1, \cdots, e_N\}$, the input is a keyword query $q$, which refers to one or more entities, and a continuous date range $t = \{d_{start}, \cdots, d_{end}\}$ where $d_{start} \leq d_{end}$, and the output is a ranked list of entities that are related to $q$, especially within $t$.*

We use DBpedia as the knowledge base in this work, which contains an enormous number of entities in different domains by extracting various kinds of structured information from Wikipedia, where each entity is tied to a Wikipedia article.

### 2.1 Probabilistic Model

We formalize the TER task as estimating the probability $P(e|q,t)$ of each entity $e$ given a keyword query $q$ and a date range $t$. The goal is then to find a ranked list of top-$k$ entities $e$, which maximize the probability $P(e|q,t)$. Based on Bayes' theorem, the probability $P(e|q,t)$ can be rewritten as follows

$$P(e|q,t) = \frac{P(e,q,t)}{P(q,t)} \propto P(e,q,t) \tag{1}$$

where the denominator $P(q,t)$ can be ignored as it does not influence the ranking.

To facilitate the discussion in the following, we first introduce the concepts of *mention* and *context*. For a keyword query $q$, a *mention* is a term in $q$ that refers to an entity $e_q$, also called *query entity*, and the *context* of $e_q$ is the set of all other mentions in $q$ except the one for $e_q$. For each query entity $e_q$, the keyword query $q$ can be decomposed into the mention and context of $e_q$, denoted by $s_{e_q}$ and $c_{e_q}$ respectively. For example, given the query entity Germany_national_football_team, the keyword query "*Germany Brazil*" results in the mention "*Germany*" and the context {"*Brazil*"}. Based on that, the joint probability $P(e,q,t)$ is given as

$$\begin{aligned} P(e,q,t) &= \sum_{e_q} P(e_q,e,q,t) = \sum_{e_q} P(e_q,e,s_{e_q},c_{e_q},t) \\ &= \sum_{e_q} P(e)P(t|e)P(e_q|e,t)P(s_{e_q}|e_q,e,t)P(c_{e_q}|e_q,e,t) \qquad (2) \\ &= \sum_{e_q} P(e)P(t|e)P(e_q|e,t)P(s_{e_q}|e_q)P(c_{e_q}|e_q,t) \qquad (3) \end{aligned}$$

where we assume in (2) $s_{e_q}$ and $c_{e_q}$ are conditionally independent given $e_q$ and $t$, in (3) $s_{e_q}$ is conditionally independent of $e$ and $t$ given $e_q$, and $c_{e_q}$ is conditionally independent of $e$ given $e_q$ and $t$. The intuition behind these assumptions is that a mention $s_{e_q}$ should only rely on the query entity $e_q$ it refers to and a context $c_{e_q}$ that appears together with $e_q$ should depend on both $e_q$ and $t$.

The main problem is then to estimate the components of $P(e, q, t)$ including the *popularity* model $P(e)$, the *temporality* model $P(t|e)$, the *relatedness* model $P(e_q|e, t)$, the *mention* model $P(s_{e_q}|e_q)$ and the *context* model $P(c_{e_q}|e_q, t)$.

### 2.2 Data Sources

To derive the estimation of these distributions in our model, we present several publicly available data sources. Based on these data sources, we discuss the details of model parameter estimation in Sec. 3.

**Wikipedia and Wikilinks.** Wikipedia provides several resources, including article titles, redirect pages and anchor text of hyperlinks, that associate each entity with terms referring to it, also called *surface forms* [11]. Wikilinks [12] also provides surface forms of entities by finding hyperlinks to Wikipedia from a Web crawl and using anchor text as mentions. Based on such sources, we construct a dictionary that maps each surface form to the corresponding entities.

Based on the observation that a more popular entity usually has more pages linking to it, we take link frequency as an indicator of *popularity*. For example, in Wikipedia the famous basketball player Michael Jeffrey Jordan is linked over 10 times more than the Berkeley professor Michael I. Jordan.

Wikipedia link structure has also been used to model *entity relatedness* [13], without considering temporal aspects, where the intuition is that Wikipedia pages containing links to both of the given entities indicate relatedness, while pages with links to only one of the given entities suggest the opposite.

**Page View Stream.** Wikipedia page view stream provides the number of times a particular Wikipedia page is requested per hour and thus can be treated as a query log of entities. In general, a well-known entity usually gets more page views than the obscure ones, such that the page view frequency also captures the *popularity* of entities.

In addition, an entity is likely to get more page views when an event related to it takes place. For example, during the FIFA World Cup, many participating football teams and players will get more page views. This explains the significant page view spike during an event when the entity receives media coverage, which has been utilized for the event detection task [14]. In this sense, the page view spike captures a user-driven measure of the *temporality* of entities.

Furthermore, an event could result in more page views for all the involved entities. For example, when Facebook acquires WhatsApp, both of them get high page view spikes. Based on this observation, simultaneous page view spikes of entities can help with modeling *the dynamic relatedness* between entities.

**Annotated Web Document Stream.** Another data source is a real-time aggregated stream of semantically annotated Web documents. We first employ a *news feed aggregator*[4] to acquire a multilingual real-time stream of news articles publicly available on the Web [15], where the enormous number of collected Web documents are in various languages, such as English (50% of all articles), German (10%), Spanish (8%) and Chinese (5%). Then we employ a cross-lingual semantic

[4] `http://newsfeed.ijs.si/visual_demo/`

annotation system[5] to annotate the multilingual Web documents with DBpedia entities, i.e., to link entity mentions to their referent entities [16]. Based on that, entity co-occurrence statistics extracted from the annotated Web documents can help to identify dynamically related entities and the co-occurrence frequency can be utilized to measure the *dynamic relatedness* between entities w.r.t. a specific time range.

### 2.3 Candidate Selection

As there are millions of entities in DBpedia, it is extremely time-consuming to calculate $P(e, q, t)$ for all entities. To improve the efficiency of TER, we employ a candidate selection process to filter out the impossible candidates. Given a query $q$ and a date range $t$, the candidate related entities are generated in three different ways: (1) Based on the dictionary containing entities and their surface forms extracted from Wikipedia and Wikilinks datasets, all query entities, whose mentions can be found in $q$, are selected as a set of candidates, denoted by $E_q$. (2) Given the set of subject, predicate and object triples $\{(s, p, o)\}$ in DBpedia, where all subjects and objects are entities, the potential candidate related entities that have a relation to the query entities are identified as $\{e|\exists p : (e, p, e_q), e_q \in E_q\} \cup \{e|\exists p : (e_q, p, e), e_q \in E_q\}$. (3) By analyzing the annotated Web documents, the entities that co-occur with the query entities in the Web documents published during the date range $t$ are also considered as candidate related entities.

## 3 Model Parameter Estimation

Our probabilistic model is parameterized by $\Phi_e = P(e)$, $\Phi_{t|e} = P(t|e)$, $\Phi_{e'|e,t} = P(e'|e, t)$, $\Phi_{s|e} = P(s|e)$ and $\Phi_{c|e,t} = P(c|e, t)$. In the following, we present the details of parameter estimation based on the introduced data sources.

### 3.1 Popularity Model $\Phi_e$

The distribution $P(e)$ captures the popularity of entity $e$. By leveraging both Wikipedia link structure and page view statistics, we first calculate $C(e)$ as

$$C(e) = C_{link}(e) + \beta C_{view}(e) \tag{4}$$

where $C_{link}(e)$ denotes the number of links pointing to $e$ and $C_{view}(e)$ denotes the average number of page views on $e$ per day. While $C_{link}(e)$ represents the prior popularity of $e$ in Wikipedia, $C_{view}(e)$ captures the popularity of $e$ based on user interests. Due to the different scales of link and page view frequencies, $C_{view}(e)$ is adjusted by a balance parameter $\beta = \frac{\text{total number of links in Wikipedia}}{\text{average number of page views per day}}$, which accounts for the difference in frequencies of Wikipedia links and per-day page views. Then the probability $P(e)$ is estimated as follows

$$P(e) = \frac{\log\left(C(e)\right) + 1}{\sum_{e_i \in W} \log\left(C(e_i)\right) + |W|} \tag{5}$$

[5] `http://km.aifb.kit.edu/sites/xlisa/`

where $W$ denotes the set of all entities. The estimation is smoothed using Laplace smoothing for avoiding the zero probability problem.

### 3.2 Temporality Model $\Phi_{t|e}$

The distribution $P(t|e)$ captures the temporality of entity $e$ w.r.t. date range $t$. We employ the page view statistics as a proxy for interest of each entity and equate the page view spike with it. For each entity $e$, we track its per-day page view counts for each date $d$. Then we compute the mean $\mu(e,d)$ and standard deviation $\sigma(e,d)$ of page views for entity $e$ in a window of $n$ days before $d$

$$\mu(e,d) = \frac{1}{n}\sum_{d_i=d-n}^{d-1} C(e,d_i) \tag{6}$$

$$\sigma(e,d) = \sqrt{\frac{1}{n}\sum_{d_i=d-n}^{d-1} (C(e,d_i)-\mu(e,d))^2} \tag{7}$$

where $C(e,d_i)$ denotes the number of page views of $e$ on date $d_i$. Inspired by [17], we calculate the page view spike $S(e,d)$ of entity $e$ on date $d$ as

$$S(e,d) = \begin{cases} \frac{C(e,d)-\mu(e,d)}{\sigma(e,d)} & \text{if } \frac{C(e,d)-\mu(e,d)}{\sigma(e,d)} \geq \kappa, \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where we assume that only the page view count $C(e,d)$ that is abnormally large compared with the previously seen page views of $e$, i.e. $\frac{C(e,d)-\mu(e,d)}{\sigma(e,d)} > \kappa$ ($\kappa$ is a fixed parameter set as 2.5 here), indicates an event and thus will be taken into account to compute the page view spike $S(e,d)$.

Based on the page view spike $S(e,d)$ of entity $e$ for date $d$, the estimation of $P(d|e)$, which is further smoothed using Laplace smoothing, is given as

$$P(d|e) = \frac{S(e,d)+\kappa}{\sum_{d_i \in T} S(e,d_i) + \kappa|T|} \tag{9}$$

where $|T|$ is the number of days contained in the longest date range $T$ supported by the system, which is set as one year here. Consequently, the probability $P(t|e)$ reflecting events about $e$ happening within $t$ can be calculated as follows (here we assume that the dates within $t$ are independent given the entity $e$)

$$P(t|e) = \sum_{d_i \in t} P(d_i|e) \tag{10}$$

### 3.3 Relatedness Model $\Phi_{e'|e,t}$

The distribution $P(e'|e,t)$ models the entity relatedness between $e$ and $e'$ w.r.t. $t$. To estimate $P(e'|e,t)$, we consider both static and dynamic entity relatedness as

$$P(e'|e,t) = \lambda \frac{R_S(e,e')}{\sum_{e'} R_S(e,e')} + (1-\lambda)\frac{R_D(e,e',t)}{\sum_{e'} R_D(e,e',t)} \tag{11}$$

where $R_S(e, e')$ measures the *static relatedness* between $e$ and $e'$, $R_D(e, e', t)$ measures the *dynamic relatedness* between $e$ and $e'$ w.r.t. $t$ and $\lambda$ is a parameter, which is set as 0.2 by default and will be discussed in detail in the experiments. For the special case that $e = e'$, we define $P(e'|e, t) = 1$.

For each pair of entities $e$ and $e'$, we calculate their *static relatedness* $R_S(e, e')$ by adopting the Wikipedia link based measure introduced by [13] as

$$R_S(e, e') = 1 - \frac{\log(\max(|E|, |E'|)) - \log(|E \cap E'|)}{\log(|W|) - \log(min(|E|, |E'|))} \tag{12}$$

where $E$ and $E'$ are the sets of entities that link to $e$ and $e'$ respectively, and $W$ is the set of all entities.

In order to measure the *dynamic relatedness* $R_D(e, e', t)$, we propose a novel approach based on *entity co-occurrence* in Web documents and *spike overlap* of page views, which will be discussed in the following.

**Entity Co-occurrence.** Based on the real-time stream of multilingual Web news articles annotated with entities, we investigate *entity co-occurrence* in the Web documents, which expresses the strength of dynamic entity association. For each pair of $e$ and $e'$ w.r.t. $t$, we calculate the entity co-occurrence measure $EC(e, e', t)$ by adopting the method of $\chi^2$ hypothesis test introduced by [18] as

$$EC(e, e', t) = \frac{N(t)(C(e, e', t)C(\overline{e}, \overline{e'}, t) - C(e, \overline{e'}, t)C(\overline{e}, e', t))^2}{C(e, t)C(e', t)(N(t) - C(e, t))(N(t) - C(e', t))} \tag{13}$$

where $N(t)$ is the total number of Web documents published within the date range $t$, $C(e, e', t)$ denotes the co-occurrence frequency of $e$ and $e'$ in the Web documents within $t$, $C(e, t)$ and $C(e', t)$ denote the frequencies of $e$ and $e'$ occurring in the Web documents within $t$, respectively, and $\overline{e}$, $\overline{e'}$ indicate that $e$, $e'$ do not occur in Web documents, i.e., $C(\overline{e}, \overline{e'}, t)$ is the number of documents within $t$ where neither $e$ nor $e'$ occurs, and $C(e, \overline{e'}, t)$ ($C(\overline{e}, e', t)$) denotes the number of documents within $t$ where $e$ ($e'$) occurs but $e'$ ($e$) does not.

**Spike Overlap.** Based on the page view spike of entities, we propose *spike overlap* $SO(e, e', t)$ to affect the dynamic relatedness between entities $e$ and $e'$ w.r.t. $t$. The intuition is that the page view spike of $e$ and $e'$ on the same date $d$ will contribute to the dynamic relatedness between $e$ and $e'$. In this regard, we calculate $SO(e, e', t)$ by adopting the weighted Jaccard similarity as

$$SO(e, e', t) = \frac{\sum_{d \in \mathcal{I}} \min\{S(e, d), S(e', d)\}}{\sum_{d \in t} \max\{S(e, d), S(e', d)\}} \tag{14}$$

where $\mathcal{I}$ can be defined as the given date range $t$, i.e., $\mathcal{I} = t$. However, the above defined measure is only based on page view spikes of entities and thus suffers from the situation that entities with significant page view spike on the same date might not be associated in reality. Therefore, we construct the date set $\mathcal{I}$ as

$$\mathcal{I} = \{d | C(e, e', d) \geq \tau, d \in t\} \tag{15}$$

where the co-occurrence frequency $C(e, e', d)$ of $e$ and $e'$ in the Web documents published on $d$ has to exceed a threshold $\tau$, which helps to determine if the page

view spike overlap is more likely to indicate an association between $e$ and $e'$ than just by chance. Based on our observation, it is reasonable to set $\tau$ as 10.

By taking both *entity co-occurrence* in Web documents and *spike overlap* of page views into consideration, we calculate the *dynamic relatedness* $R_D(e, e', t)$ between entities $e$ and $e'$ for a specific date range $t$ as follows

$$R_D(e, e', t) = EC(e, e', t) \cdot SO(e, e', t)^2 \tag{16}$$

### 3.4 Mention Model $\Phi_{s|e}$

The distribution $P(s|e)$ models the likelihood of observing the mention $s$ given the intended entity $e$. To estimate $P(s|e)$, we employ Wikipedia and Wikilinks datasets and propose a point-wise mutual information (PMI) based method as

$$P(s|e) = \frac{PMI(e, s)}{\sum_{s_i \in S_e} PMI(e, s_i)} \tag{17}$$

where $S_e$ is the set of surface forms of entity $e$ and $PMI(e, s)$ is calculated as

$$PMI(s, e) = \log \frac{P(s, e)}{P(s)P(e)} = \log \frac{C(e, s) \times N}{C(s) \times C(e)} \tag{18}$$

where we have $P(s) = \frac{C(s)}{N}$, $P(e) = \frac{C(e)}{N}$, $P(s, e) = \frac{C(e,s)}{N}$ based on maximum likelihood estimation (MLE), $C(s)$ is the number of links using $s$ as anchor text, $C(e)$ is the number of links pointing to $e$, $C(e, s)$ is the number of links using $s$ as anchor text pointing to $e$ and $N$ is the total number of links.

### 3.5 Context Model $\Phi_{c|e,t}$

The probability $P(c|e, t)$ models the likelihood of observing the context $c$ given the query entity $e$ and the date range $t$. The context $c$ of $e$ contains the surface forms of other entities related to $e$. Assuming that all surface forms $s_c$ in the context $c$ are independent given $e$ and $t$, the probability $P(c|e, t)$ is estimated as

$$P(c|e, t) = \prod_{s_c \in c} P(s_c|e, t) \tag{19}$$

The problem remains to estimate $P(s_c|e, t)$, the probability that a surface form $s_c$ appears in the context of $e$ w.r.t. $t$.

Given the query entity $e$ and date range $t$, we consider a generation process of the context, where the context model first finds the *related entities* of $e$ w.r.t. $t$ based on the relatedness model, and then generates the surface form $s_c$ of such related entities as the context of $e$ based on the mention model. The form of the context generation for the query entity $e$ and date range $t$ is given as

$$P_R(s_c|e, t) = \sum_{e_{s_c} \in E_{s_c}} P(e_{s_c}, s_c|e, t) = \sum_{e_{s_c} \in E_{s_c}} \underbrace{P(e_{s_c}|e, t)}_{\textbf{Relatedness}} \underbrace{P(s_c|e_{s_c})}_{\textbf{Mention}} \tag{20}$$

where $E_{s_c}$ denotes the set of entities having surface form $s_c$ and we assume that $s_c$ is independent of $e$ and $t$ given $e_{s_c}$, i.e., $P(s_c|e_{s_c}, e, t) = P(s_c|e_{s_c})$.

The above estimation suffers from the sparse data problem, i.e., some entities are not related to a given query entity $e$, but might appear as the context of $e$ in the query $q$, which results in zero probability. Therefore, we perform smoothing by giving some probability mass to such unrelated entities. The general idea is that a surface form $s_c$ of entities that are not related to the query entity $e$ should also be possible to appear in the context of $e$ and can be generated by chance. In this regard, we define the probability $P(s)$ of surface form $s$, which is built from the *entire collection* of entities and surface forms, as

$$P(s) = \frac{\sum_{e \in E_s} C(e, s)}{\sum_{s_i \in S} \sum_{e_i \in E_{s_i}} C(e_i, s_i)} \tag{21}$$

where $S$ is the set of all surface forms, $E_s$ is the set of entities having surface form $s$, and $C(e, s)$ denotes the frequency that $s$ refers to $e$.

In order to achieve a robust estimation of the context model, we further smooth $P_R(s_c|e, t)$ using $P(s)$ based on Jelinek-Mercer smoothing as follows

$$P(s_c|e, t) = \gamma P_R(s_c|e, t) + (1 - \gamma) P(s_c) \tag{22}$$

where $\gamma$ is a tunable parameter that is set to 0.9 by line search in our experiments. This estimation mixes the probability of $s_c$ derived from the related entities of $e$ with the general collection frequency of $s_c$ used to refer to any entities.

## 4 Evaluation

We now discuss the experiments we have conducted to assess the performance of our approach to TER based on our newly created benchmark datasets.

### 4.1 Experimental Setup

In our experiments, we employ DBpedia 2014[6] as the knowledge base and the Wikipedia snapshot of June 2014 as the auxiliary data source. Existing datasets for the evaluation of entity recommendation aim to quantify the degree to which entities are related to the query without involving temporal aspects, which makes such datasets unsuitable for the TER task. There are some studies using a subset of TREC queries for time-aware information retrieval, where the goal is to investigate the user's implicit temporal intent for document retrieval [19,20]. However, such datasets do not contain the time ranges of interest explicitly given by users along with the queries and thus cannot be used for the TER evaluation. Therefore, we have created a new dataset where we asked 6 volunteers, who also serve as judges of the experimental results, to provide information needs of both queries and date ranges. By removing the duplicate ones, it results in a final set

[6] `http://wiki.dbpedia.org/Downloads2014`

of 22 information needs in different domains including Sports, Entertainment, Business, Emergencies, Society, Science and Politics. The datasets used in our experiments are available at `http://km.aifb.kit.edu/sites/ter/`.

To the best of our knowledge, no existing work on the TER task can be found. Therefore, we build the following baselines for comparison with our approach: (1) the first baseline is a static method using an ad hoc ranking function without considering the given time range $t$, defined as $Score(e,q) = \sum_{e_q} C(e_q) R_S(e_q, e)$, where $C(e_q)$ represents the commonness of each query entity $e_q$ w.r.t. the corresponding mention in the query $q$, which has been introduced by [21,11], and $R_S(e_q, e)$ denotes the Wikipedia link based relatedness between each query entity $e_q$ and the candidate entity $e$ [13]; (2) the second baseline is similar to our probabilistic model, but without taking into account the time range $t$, defined as $P(e,q) = \sum_{e_q} P(e)P(e_q|e)P(s_{e_q}|e_q)P(c_{e_q}|e_q)$, where $P(e)$ and $P(s_{e_q}|e_q)$ are estimated using our popularity and mention models respectively, $P(e_q|e)$ and $P(c_{e_q}|e_q)$ are also estimated using our relatedness and context models, but with $\lambda = 1$ (see Eq. 11), i.e., only the static relatedness between entities is considered in these models. For a comparative analysis, we have conducted the experiments with several methods: the above described two baselines, denoted by *BSL1* and *BSL2*, respectively; our proposed method leaving out each of the popularity, temporality, relatedness, mention and context models, denoted by $-\Phi_e$, $-\Phi_{t|e}$, $-\Phi_{e'|e,t}$, $-\Phi_{s|e}$ and $-\Phi_{c|e,t}$, respectively; and our method with all these five models, denoted by *Full Model*.

The existing work, such as the Spark system from Yahoo! [4] and the similar one published by Microsoft [9,10], could also be used for comparison with our method, even though they are not dedicated to the TER task. However, these systems assume that a query refers to only one entity, so they cannot deal with our more general case, where the query could involve multiple query entities. More importantly, these systems rely on the datasets that only major Web search engines have and are not publicly accessible. Due to these reasons, it is difficult to re-implement such systems and compare them with our method.

### 4.2 Results of Entity Retrieval

To assess the quality of entities retrieved by our method, we employ Normalized Discounted Cumulative Gain (nDCG) at rank $k$ [22] as quality criteria, which is defined as $nDCG@k = \frac{DCG@k}{IDCG@k}$, where $DCG@k = \sum_{i=0}^{k} \frac{2^{rel_i}-1}{log_2(i+1)}$ and $rel_i$ is the graded relevance assigned to the result at position $i$ and $IDCG@k$ is the maximum attainable $DCG@k$. This measure captures the goodness of a retrieval model based on the graded relevance of the top-$k$ results. For each information need, all the entities retrieved by different methods are judged on 1-5 relevance scale by the 6 volunteers based on the criteria including both relevance and timeliness w.r.t. the underlying information needs. The final relevance of each candidate entity is determined by the relevance score voted by most judges and ties are resolved by the authors. More details about the description of each graded relevance are available in our datasets.

| | nDCG@k | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $BSL1$ | $BSL2$ | $-\Phi_e$ | $-\Phi_{t\|e}$ | $-\Phi_{e'\|e,t}$ | $-\Phi_{s\|e}$ | $-\Phi_{c\|e,t}$ | ***Full Model*** |
| *k=5* | 0.597 | 0.622 | 0.805 | 0.778 | 0.140 | 0.800 | 0.797 | **0.824** |
| *k=10* | 0.594 | 0.621 | 0.817 | 0.786 | 0.176 | 0.803 | 0.804 | **0.839** |
| *k=15* | 0.596 | 0.640 | 0.846 | 0.810 | 0.505 | 0.830 | 0.823 | **0.859** |
| *k=20* | 0.616 | 0.642 | 0.865 | 0.831 | 0.521 | 0.853 | 0.847 | **0.879** |
| *k=30* | 0.635 | 0.658 | 0.898 | 0.877 | 0.552 | 0.895 | 0.887 | **0.925** |

Table 1: $nDCG@k$ of retrieved entities (with the best results in bold).

| | Recall@k | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | $BSL1$ | $BSL2$ | $-\Phi_e$ | $-\Phi_{t\|e}$ | $-\Phi_{e'\|e,t}$ | $-\Phi_{s\|e}$ | $-\Phi_{c\|e,t}$ | ***Full Model*** |
| *k=5* | 0.273 | 0.264 | 0.464 | 0.464 | 0.091 | 0.491 | 0.491 | **0.518** |
| *k=10* | 0.318 | 0.309 | 0.582 | 0.591 | 0.146 | 0.591 | 0.600 | **0.646** |
| *k=15* | 0.318 | 0.336 | 0.655 | 0.655 | 0.182 | 0.700 | 0.700 | **0.736** |
| *k=20* | 0.346 | 0.346 | 0.709 | 0.682 | 0.255 | 0.746 | 0.736 | **0.755** |
| *k=30* | 0.364 | 0.364 | 0.791 | 0.827 | 0.318 | 0.846 | 0.809 | **0.855** |

Table 2: $Recall@k$ of temporally related entities (with the best results in bold).

The experimental results of $nDCG@k$ with varying $k$ for different methods are shown in Table 1. Our method with *Full Model* performs the best for different $k$. Compared with the static baseline $BSL2$ using a similar probabilistic model, it achieves 32.5%, 35.1%, 34.2%, 36.9% and 40.6% improvements when $k$ is 5, 10, 15, 20 and 30, respectively. The baselines only obtain better results compared with our method without the relatedness model, while our method leaving out any other model still greatly outperforms the baselines. By comparing the two static baselines, *BSL2* clearly outperforms *BSL1*, which also shows the advantage of the method based on our probabilistic model over the ad hoc method.

As we focus on the TER task, the capability of our method to find temporally related entities is of great importance such that we have created an additional dataset consisting of only temporally related entities, which are also determined based on the votes of the 6 judges. Firstly, they are asked to select the entities that are temporally related to each information need and such entities are then ranked by the number of times being selected. Only the top-5 ranked candidates are included into the final dataset, where ties are resolved by the authors. This results in 110 entities in total (5 for each of the 22 information needs).

In this experimental setting, we are concerned with whether these temporally related entities can appear on top of the ranked list of the retrieved entities. For this, we consider recall at rank $k$ ($recall@k$) as quality criteria, where recall defines the number of relevant results that are retrieved in relation to the total number of relevant results and recall@$k$ is defined by only taking into account the top-$k$ results. The experimental results of $recall@k$ with varying $k$ for different methods are shown in Table 2. While the two static baselines exhibit only minor differences, our method with *Full Model* achieves a considerable performance improvement over the baselines for different $k$.

For both measures of $nDCG@k$ and $recall@k$, we observe that our method achieves better results by adding each individual model and the relatedness model that incorporates both static and dynamic entity relatedness contributes the most. For example, when $k = 30$, $nDCG@k$ and $recall@k$ decrease 40.1% and 62.8% respectively, by ablating the relatedness model, while the performance reduction without the other models ranges from 5.2% to 2.9% for $nDCG@k$ and from 7.5% to 1.1% for $recall@k$.

| *Gold Standard* | *BSL2* | *Full Model* |
|---|---|---|
| **Germany nat'l football team** | Latin America | **Brazil nat'l football team** |
| **Brazil nat'l football team** | **Brazil nat'l football team** | **Germany nat'l football team** |
| **2014 FIFA World Cup** | Brazil nat'l basketball team | **2014 FIFA World Cup** |
| **Joachim Löw** | **2014 FIFA World Cup** | **Luiz Felipe Scolar**i |
| **Toni Kroos** | **Germany nat'l football team** | FIFA World Rankings |
| **Luiz Felipe Scolari** | FIFA World Rankings | **Toni Kroos** |
| **Neymar** | **Luiz Felipe Scolari** | **Neymar** |
| FIFA World Rankings | **Neymar** | **Joachim Löw** |
| Latin America | **Joachim Löw** | Latin America |
| Brazil nat'l basketball team | **Toni Kroos** | Brazil nat'l basketball team |

Table 3: The gold-standard ranking of 10 entities (with dynamically related ones in bold) for the query *"Germany Brazil"* and the date range *"July 2014"* as well as the rankings by the baseline *BSL2* and our method with *Full Model*.

| **Domain (#Query)** | $BSL1$ | $BSL2$ | $-\Phi_e$ | $-\Phi_{t\|e}$ | $-\Phi_{e'\|e,t}$ | $-\Phi_{s\|e}$ | $-\Phi_{c\|e,t}$ | ***Full Model*** |
|---|---|---|---|---|---|---|---|---|
| *Sports (6)* | 0.149 | 0.289 | 0.531 | 0.572 | 0.240 | 0.646 | 0.529 | **0.663** |
| *Entertainment (4)* | 0.191 | 0.252 | 0.594 | 0.645 | 0.188 | 0.667 | 0.673 | **0.688** |
| *Business (3)* | 0.596 | 0.596 | 0.790 | 0.834 | -0.139 | 0.838 | **0.855** | 0.838 |
| *Emergencies (4)* | -0.130 | -0.082 | 0.473 | 0.421 | 0.470 | 0.503 | 0.467 | **0.494** |
| *Others (5)* | 0.365 | 0.358 | **0.612** | 0.522 | 0.232 | 0.576 | 0.527 | 0.581 |
| **Average** | 0.216 | 0.272 | 0.586 | 0.582 | 0.219 | 0.634 | 0.588 | **0.642** |

Table 4: Spearman rank correlation between the gold-standard ranking and the ranking generated by different methods (with the best results in bold).

### 4.3 Results of Entity Ranking

The measures of nDCG@$k$ and recall@$k$ assess the quality of only top-$k$ results, while we would like to evaluate the ranking of entities from highly relevant ones to only remotely relevant or even not relevant ones. Therefore, we have created another dataset, where the authors select 10 candidate entities for each information need in a way that their relevances are clearly distinguishable among each other. Similar to [23], the gold-standard ranking of the 10 candidate entities is then created in the following way: (1) for all possible comparisons of the 10 candidate entities (45 in total), the 6 judges are asked which of the given two entities is more related to the information need by considering both relevance and timeliness; (2) all comparisons are then aggregated into a single confidence value for each entity and the 10 candidate entities are ranked by these confidence values as described by [24]. The final output is a set of 22 ranked lists consisting of 10 entities for each, against which we compare the automatically generated rankings by different methods using Spearman rank correlation, which measures the strength of association between two ranked variables. Some examples of different rankings are shown in Table 3.

The Spearman rank correlation between the gold-standard ranking and the automatically generated rankings by all these methods is given in Table 4. It shows that the experimental results of entity ranking are consistent with the results obtained in the entity retrieval experiments. The static baseline *BSL2* with a probabilistic model yields slightly better results than the baseline *BSL1* that is based on an ad hoc method. Clearly, our method with *Full Model* achieves the best results and considerably outperforms the baselines. Similarly, all the individual models contribute to the final performance improvement, where the relatedness model contributes the most. By respectively ablating the models $\Phi_e$,

| **Domain** | $\lambda = 0$ | $\lambda = .1$ | $\lambda = .2$ | $\lambda = .3$ | $\lambda = .4$ | $\lambda = .5$ | $\lambda = .6$ | $\lambda = .7$ | $\lambda = .8$ | $\lambda = .9$ | $\lambda = 1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| *Sports* | 0.620 | 0.653 | **0.663** | 0.636 | 0.634 | 0.610 | 0.604 | 0.564 | 0.541 | 0.489 | 0.285 |
| *Entertainment* | 0.573 | 0.670 | **0.688** | 0.636 | 0.612 | 0.530 | 0.512 | 0.473 | 0.445 | 0.439 | 0.348 |
| *Business* | 0.737 | 0.838 | 0.838 | **0.842** | **0.842** | **0.842** | 0.822 | 0.826 | 0.834 | 0.794 | 0.657 |
| *Emergencies* | **0.530** | 0.518 | 0.494 | 0.509 | 0.467 | 0.479 | 0.458 | 0.412 | 0.367 | 0.303 | -0.058 |
| *Others* | 0.537 | 0.576 | **0.581** | 0.564 | 0.537 | 0.503 | 0.505 | 0.534 | 0.537 | 0.493 | 0.280 |
| **Average** | 0.592 | 0.639 | **0.642** | 0.625 | 0.606 | 0.579 | 0.568 | 0.549 | 0.531 | 0.489 | 0.284 |

Table 5: Spearman rank correlation between the gold-standard ranking and the ranking by our *Full Model* for different $\lambda$ (with the best results in bold).

$\Phi_{t|e}$, $\Phi_{e'|e,t}$, $\Phi_{s|e}$ and $\Phi_{c|e,t}$, the performance correspondingly reduces 8.7%, 9.3%, 65.8%, 1.2% and 8.4%.

Our method is sensitive to the parameter $\lambda$ used in the relatedness model (see Eq. 11). Intuitively, a smaller $\lambda$ reflects that the dynamic entity relatedness measure plays a more important role in the model. Table 5 shows the impact of $\lambda$ on the ranking performance of our method with *Full Model*, where $\lambda = 0.2$ yields the best results on average, which has been used as the default value in our experiments. We observe that only using the dynamic relatedness measure, i.e., $\lambda = 0$, achieves the best results for the *Emergencies* domain. This is because in this domain there are more entities that are only dynamically related to the query. For example, given the information need about the crash of Indonesia AirAsia Flight 8501 into the Java sea in December 2014, where the query is *"Indonesia Java"* and the date range is *"December 2014"*, the related entities AirAsia, Aviation_accidents_and_incidents and Search_and_rescue do not have a static connection with the query. Another tunable parameter is $\gamma$ (see Eq. 22). We observe that $\gamma = 0.9$ achieves the best results, which has been set as the default value in our experiments. For the sake of space, we omit the results based on different $\gamma$ because they exhibit only minor differences.

## 5 Related Work

The TER task can be placed in the context of (1) entity search, (2) related entity recommendation and (3) temporal information retrieval.

*Entity search* has been defined by [1] as finding entities explicitly named in the query. Recently, entity search becomes more complex and closer to question answering when the query only provides a description of the target entity, where a list of member relationships to a single entity is given in the query. A recent development in evaluating entity search of this type was the introduction of the Related Entity Finding using Linked Open Data (REF-LOD) task at the TREC Entity Track in 2010 and 2011 [25], where the type of relation to the target entity and the type of the target entity are both given as constraints.

For *related entity recommendation*, the Spark system developed at Yahoo! extracts several features from a variety of data sources and uses a machine learning model to produce a recommendation of entities to a Web search query, where neither the relation type nor the type of the target entity are specified [4]. Following Spark, Sundog aims to improve entity recommendation, in particular with respect to freshness, by exploiting Web search log data using a stream processing based implementation [8]. Microsoft has also developed a similar

system that performs personalized entity recommendation by analyzing user click logs and entity pane logs [9,10].

In recent years, the time dimension has received a large share of attention in *temporal information retrieval* [5]. The temporal characteristics of queries [26] and dynamics of document content [27] have been leveraged in relevance ranking. The real-time information extracted from Twitter has been used to train learning to rank models [28]. To improve Web search results, the temporal information has also been used for query understanding [29] and auto-completion of queries [30].

## 6 Conclusions

In this paper, we introduce a novel task of *time-aware entity recommendation* (TER), since we argue that time-awareness should be a crucial factor in entity recommendation, which has not been addressed so far. To tackle this challenge, we propose a probabilistic model that aims to rank related entities according to a time-specific information need presented as a keyword query and a date range. The main contribution of our approach is that we decompose the TER task into several well defined probability distributions, each representing the context of a different component in the model. Through these components, heterogeneous entity knowledge extracted from different data sources that are publicly available on the Web can be incorporated into our model. Experimental results show that our method clearly outperforms approaches that are not context-aware, specifically when being time-agnostic.

**Acknowledgments.** The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 611346.

## References

1. Pound, J., Mika, P., Zaragoza, H.: Ad-hoc object retrieval in the web of data. In: WWW. (2010) 771–780
2. van Zwol, R., Pueyo, L.G., Muralidharan, M., Sigurbjörnsson, B.: Machine learned ranking of entity facets. In: SIGIR. (2010) 879–880
3. Kang, C., Vadrevu, S., Zhang, R., van Zwol, R., Pueyo, L.G., Torzec, N., He, J., Chang, Y.: Ranking related entities for web search queries. In: WWW (Companion Volume). (2011) 67–68
4. Blanco, R., Cambazoglu, B.B., Mika, P., Torzec, N.: Entity recommendations in web search. In: ISWC. (2013) 33–48
5. Kanhabua, N., Blanco, R., Nørvåg, K.: Temporal information retrieval. Foundations and Trends in Information Retrieval **9**(2) (2015) 91–208
6. Nørvåg, K.: Supporting temporal text-containment queries in temporal document databases. Data Knowl. Eng. **49**(1) (2004) 105–125
7. Berberich, K., Bedathur, S.J., Neumann, T., Weikum, G.: A time machine for text search. In: SIGIR. (2007) 519–526

8. Fischer, L., Blanco, R., Mika, P., Bernstein, A.: Timely semantics: A study of a stream-based ranking system for entity relationships. In: ISWC. (2015) 429–445
9. Yu, X., Ma, H., Hsu, B.P., Han, J.: On building entity recommender systems using user click log and freebase knowledge. In: WSDM. (2014) 263–272
10. Bi, B., Ma, H., Hsu, B.P., Chu, W., Wang, K., Cho, J.: Learning to recommend related entities to search users. In: WSDM. (2015) 139–148
11. Shen, W., Wang, J., Luo, P., Wang, M.: LINDEN: linking named entities with knowledge base via semantic knowledge. In: WWW. (2012) 449–458
12. Singh, S., Subramanya, A., Pereira, F., McCallum, A.: Wikilinks: A large-scale cross-document coreference corpus labeled via links to Wikipedia. Technical Report UM-CS-2012-015 (2012)
13. Milne, D., Witten, I.H.: An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In: AAAI Workshop on Wikipedia and Artificial Intelligence. (2008)
14. Ciglan, M., Nørvåg, K.: Wikipop: personalized event detection system based on wikipedia page view statistics. In: CIKM. (2010) 1931–1932
15. Trampuš, M., Novak, B.: Internals of an aggregated web news feed. In: SiKDD. (2012) 431–434
16. Zhang, L., Rettinger, A.: X-lisa: Cross-lingual semantic annotation. PVLDB **7**(13) (2014) 1693–1696
17. Osborne, M., Petrovic, S., McCreadie, R., Macdonald, C., Ounis, I.: Bieber no more: First Story Detection using Twitter and Wikipedia. In: SIGIR 2012 Workshop on Time-aware Information Access. (2012)
18. Bron, M., Balog, K., de Rijke, M.: Ranking related entities: components and analyses. In: CIKM. (2010) 1079–1088
19. Li, X., Croft, W.B.: Time-based language models. In: CIKM. (2003) 469–475
20. Kanhabua, N., Nørvåg, K.: Determining time of queries for re-ranking search results. In: ECDL. (2010) 261–272
21. Milne, D.N., Witten, I.H.: Learning to link with wikipedia. In: CIKM. (2008) 509–518
22. Järvelin, K., Kekäläinen, J.: IR evaluation methods for retrieving highly relevant documents. In: SIGIR. (2000) 41–48
23. Hoffart, J., Seufert, S., Nguyen, D.B., Theobald, M., Weikum, G.: KORE: keyphrase overlap relatedness for entity disambiguation. In: CIKM. (2012) 545–554
24. Coppersmith, D., Fleischer, L., Rudra, A.: Ordering by weighted number of wins gives a good ranking for weighted tournaments. ACM Transactions on Algorithms **6**(3) (2010)
25. Balog, K., Serdyukov, P., de Vries, A.P.: Overview of the TREC 2011 entity track. In: TREC. (2011)
26. Dai, N., Davison, B.D.: Freshness matters: in flowers, food, and web authority. In: SIGIR. (2010) 114–121
27. Elsas, J.L., Dumais, S.T.: Leveraging temporal dynamics of document content in relevance ranking. In: WSDM. (2010) 1–10
28. Dong, A., Zhang, R., Kolari, P., Bai, J., Diaz, F., Chang, Y., Zheng, Z., Zha, H.: Time is of the essence: improving recency ranking using twitter data. In: WWW. (2010) 331–340
29. Kulkarni, A., Teevan, J., Svore, K.M., Dumais, S.T.: Understanding temporal query dynamics. In: WSDM. (2011) 167–176
30. Shokouhi, M., Radinsky, K.: Time-sensitive query auto-completion. In: SIGIR. (2012) 601–610