Inf Retrieval J (2017) 20:363–398
DOI 10.1007/s10791-017-9302-1

Identifying top relevant dates for implicit time sensitive queries

Ricardo Campos[1,2] · **Gaël Dias**[3] · **Alípio Mário Jorge**[2,4] · **Célia Nunes**[5,6]

Received: 23 July 2015 / Accepted: 21 April 2017 / Published online: 5 May 2017

Abstract Despite a clear improvement of search and retrieval temporal applications, current search engines are still mostly unaware of the temporal dimension. Indeed, in most cases, systems are limited to offering the user the chance to restrict the search to a particular time period or to simply rely on an explicitly specified time span. If the user is not explicit in his/her search intents (e.g., "*philip seymour hoffman*") search engines may likely fail to present an overall historic perspective of the topic. In most such cases, they are limited to retrieving the most recent results. One possible solution to this shortcoming is to understand the different time periods of the query. In this context, most state-of-the-art methodologies consider any occurrence of temporal expressions in web documents and other web data as equally relevant to an implicit time sensitive query. To approach this problem in a more adequate manner, we propose in this paper the detection of relevant temporal expressions to the query. Unlike previous metadata and query log-based

A short version of this work has appeared in Proceedings of TempWeb@WWW'12 (Campos et al. 2012).

✉ Ricardo Campos
ricardo.campos@ipt.pt; ricardo.campos@inesctec.pt

Gaël Dias
gael.dias@unicaen.fr

Alípio Mário Jorge
amjorge@fc.up.pt

Célia Nunes
celian@ubi.pt

[1] ICT Department, Polytechnic Institute of Tomar, Tomar, Portugal

[2] LIAAD/INESC TEC - INESC Technology and Science, Porto, Portugal

[3] HULTECH/GREYC, University of Caen Basse-Normandie, Caen, France

[4] DCC – Faculty of Sciences, University of Porto, Porto, Portugal

[5] Department of Mathematics, University of Beira Interior, Covilhã, Portugal

[6] Center of Mathematics and Applications, University of Beira Interior, Covilhã, Portugal

approaches, we show how to achieve this goal based on information extracted from document content. However, instead of simply focusing on the detection of the most obvious date we are also interested in retrieving the set of dates that are relevant to the query. Towards this goal, we define a general similarity measure that makes use of co-occurrences of words and years based on corpus statistics and a classification methodology that is able to identify the set of top relevant dates for a given implicit time sensitive query, while filtering out the non-relevant ones. Through extensive experimental evaluation, we mean to demonstrate that our approach offers promising results in the field of temporal information retrieval (T-IR), as demonstrated by the experiments conducted over several baselines on web corpora collections.

Keywords Temporal information retrieval · Implicit time sensitive queries · Temporal query understanding · Relevant temporal expressions

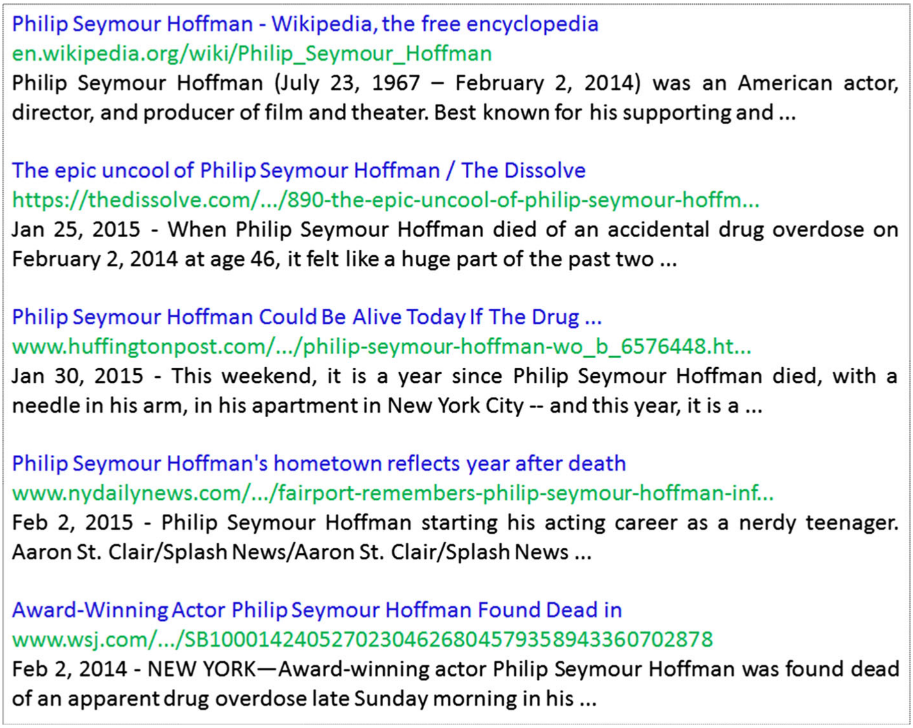
1 Introduction

Search engines typically return a ranked list of documents in response to a user's query. For each document, its title, snippet and URL are usually presented. This information is usually very useful to the user as it helps to decide which of the results are of interest. Finding the required information may, however, be a difficult task especially when users are not explicit in their search intents. Such a problem, related to time, usually arises when users are silent with regards to their temporal information needs, issuing implicit temporal queries (e.g., "*fukushima*") instead of explicit temporal ones (e.g., "*fukushima 2011*"). A very significant problem in that most of the temporal queries issued by users are implicit by nature (Campos et al. 2011b; Metzler et al. 2009; Nunes et al. 2008). For such temporal queries, search engines usually push to the top the most recent search results based on the document timestamp as it can be seen in Fig. 1. This may work well for queries going after recent results, e.g., "*philip seymour hoffman"* at the time of his death, but it may be of little relevance to users interested in more fine-grained time information, both at that time or later.

A more elaborate mechanism should give users the possibility to specify a point-in-time or a temporal interval in order to filter the results. However, shifting the burden of temporally tagging the query from computers to the users is simply unwise. Besides, users do not always know which time interval to specify.

Overcoming these problems demands search systems to automatically determine the set of relevant years that are related to an implicit temporal query. Our research hypothesis is that the introduction of a classification model that is able to identify top relevant dates for any given implicit time sensitive query while filtering out non-relevant ones, improves the correct classification of a query and a candidate date pair when compared to those approaches which consider all the candidate dates as relevant for the query.

To tackle this problem, we adopt a web content analysis approach that extracts temporal expressions from web documents. The high number of temporal expressions that can be found on this type of collection poses, however, some challenges since only a few of them are actually relevant to the query. Hence, our goal is twofold: (1) to select the most relevant dates for a given query and (2) to discard all non-relevant or incorrect ones. In order to accomplish our objectives, we adopt a twofold approach: (1) firstly, we present our Generic



Philip Seymour Hoffman - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Philip_Seymour_Hoffman
Philip Seymour Hoffman (July 23, 1967 – February 2, 2014) was an American actor, director, and producer of film and theater. Best known for his supporting and ...

The epic uncool of Philip Seymour Hoffman / The Dissolve
https://thedissolve.com/.../890-the-epic-uncool-of-philip-seymour-hoffm...
Jan 25, 2015 - When Philip Seymour Hoffman died of an accidental drug overdose on February 2, 2014 at age 46, it felt like a huge part of the past two ...

Philip Seymour Hoffman Could Be Alive Today If The Drug ...
www.huffingtonpost.com/.../philip-seymour-hoffman-wo_b_6576448.ht...
Jan 30, 2015 - This weekend, it is a year since Philip Seymour Hoffman died, with a needle in his arm, in his apartment in New York City -- and this year, it is a ...

Philip Seymour Hoffman's hometown reflects year after death
www.nydailynews.com/.../fairport-remembers-philip-seymour-hoffman-inf...
Feb 2, 2015 - Philip Seymour Hoffman starting his acting career as a nerdy teenager. Aaron St. Clair/Splash News/Aaron St. Clair/Splash News ...

Award-Winning Actor Philip Seymour Hoffman Found Dead in
www.wsj.com/.../SB10001424052702304626804579358943360702878
Feb 2, 2014 - NEW YORK—Award-winning actor Philip Seymour Hoffman was found dead of an apparent drug overdose late Sunday morning in his ...

Fig. 1 Top-5 Google results for the query "*philip seymour hoffman*" on June 5, 2015

Temporal Evaluation measure (GTE) which evaluates the temporal similarity, i.e., the temporal relatedness between a query and a candidate date; and (2) secondly, we propose a classification model (GTE-Class) to accurately relate relevant dates to their corresponding query terms and filter out non-relevant ones. The effectiveness of our approach is assessed using several testbeds including a web test collection based on TREC queries (from TREC2013-ts and TREC2014-ts task).

Our contributions can be summarized as follows: (1) we propose a novel approach to tag text queries with relevant temporal expressions by relying on a content-based approach and a classification methodology; (2) our generic temporal similarity measure, GTE, outperforms both well-known first order similarity measures, as well as state-of-the-art approaches; (3) our date filtering approach, GTE-Class, is able to achieve better results when compared to state-of-the-art machine learning approaches and (4) we make available to the scientific community a set of real-world queries and ground-truth results, fostering the development and the assessment of future approaches. This manuscript is an extended version of Campos et al. (2012) presented at the TempWeb@WWW'12 workshop. In comparison with that work, this article:

- Gives a high-level overview of the related research on temporal information retrieval (T-IR);
- Details the algorithm employed while using a running example, including significant extension in terms of parameter settings;

- Extensively highlights the experimental part of the work with a whole new set of comparative experiments with four additional state-of-the-art baseline approaches that make use of temporal signals: Kanhabua and Nørvåg (2010); Strötgen et al. (2012) for GTE; Kanhabua et al. (2012) and Kawai et al. (2010) for GTE-Class.
- Incorporates an additional test collection to further corroborate that the effectiveness of our work is not limited to a single dataset.

The remainder of this article is structured as follows. Section 2 gives a detail-rich contextualization of the motivation behind our work. Section 3 offers an overview of related research. Section 4 defines both the GTE and the classification methodology (GTE-Class). Section 5 describes the experimental setup and discusses the obtained results. Finally, Sect. 6 summarizes the article and ends with some final remarks presenting research avenues that can be explored in the future.

2 Motivation

With an increasing number of collections and the spread of information, finding the correct answer turns out to be a difficult task for any user searching on the web, especially when it comes to temporal-dependent information needs. Consider, for instance, a user with an information need on the explosion and sinking of the Deepwater Horizon oil rig in the Gulf of Mexico, who might issue the query "*bp oil spill*". It might be possible that the user is interested in information referring to the very concrete point in time (2010) of this disaster. But, this user might also be interested in many other subsequent details related to the event, such as the year 2011 when the BP operations ceased, 2014 when BP was appointed by a court as the primarily responsible for the oil spill, or more recently 2015 when BP agreed to pay $18.7 billion in fines. Another example is a user who has an interest on "*haiti earthquakes*", an event that may refer to two different points in time, i.e., 1564 and 2010, but also to the year 2011, as the one year anniversary of the *2010 haiti earthquake*.

In general, most of the systems will be simply interested in determining and presenting information about the most obvious point in time. However, users with longitudinal information needs might also be interested in countless other related aspects, prior or subsequent to the main event. In this work, we are particularly interested in retrieving not only the obvious points in time but also the set of correlated dates. We specifically put the focus on (1) temporally unambiguous queries, i.e., queries taking place in a very concrete time period (e.g., "*bp oil spill*") and on (2) temporally ambiguous ones, i.e., queries that have multiple instances over time (e.g., "*haiti earthquake*"). We believe that finding the correct time points associated to a query is critical to improve search retrieval systems in a number of temporal related applications that range from clustering, query expansion, event tracking or re-ranking search results.

As a concrete example, consider a user seeking a digital library or a digitized book (Foley and Allan 2015). To retrieve information about the French mathematician, physicist, inventor, writer and Christian philosopher Blaise Pascal, the user issues the query "*Blaise Pascal*" to express that specific information need. Any information related to Pascal's birth date and posterior, including his calculators and later treatises on several subjects, would be of interest to the user.

Now imagine a user looking for an automatic summarization of news topics over time (Tran et al. 2013; Tran et al. 2015). For example, a timeline of "*Donald Trump election*" would be of interest to someone looking for more specific, though balanced information on this matter. In both scenarios, one search system able to find the relevant dates for the queries might favor documents close to the query related dates, therefore avoiding the task of searching to become into a burden and distressful one.

Finding the correct dates of a query can also be useful to improve the IR system itself, for instance, as a strong important dimension that improves the effectiveness of document ranking search results, such that relevant documents combining a keyword and a temporal natural get promoted to the top (Campos et al. 2016; Brucato and Danilo 2014; Campos et al. 2014c; Kanhabua and Nørvåg 2010). The need for better contextualized information, also demands search systems to explore new forms of presenting the information. Finding the relevant dates of a query and presenting them by means of a cluster (Alonso et al. 2009b; Campos et al. 2014b) can also be very useful for users seeking for information in devices with a small screen. This visualization approach will enable users to get an overall perspective of a given topic.

Taking into consideration that information about time is becoming increasingly important and is not limited to finding the relevant dates for a query. For example, another strand of research standing at the crossroads between information retrieval and time is the correct estimation of a document's publication date (Jatowt et al. 2013; Kanhabua and Nørvåg 2008). State of the art approaches are often based on determining the intent of user queries (Radinsky et al. 2011; Ren et al. 2013). Other works, involve understanding the dynamics of temporal queries (Kulkarni et al. 2011). A more thorough discussion of the related work on temporal and information retrieval applications can be found in a number of surveys (Campos et al. 2014a; Kanhabua et al. 2015; Moulahi et al. 2015) that were recently published on this matter.

All these examples highlight the importance of finding relevant dates to the query when those dates are simply missing. This motivates our work to develop a measure that addresses this problem.

3 Related research

The process of searching for information is inherently temporal. Even though some user information needs may be explicitly expressed, most are implicit by nature (Campos et al. 2011b; Metzler et al. 2009; Nunes et al. 2008). However, determining the user's temporal intent underlying a given query is a tough task. In this section, we provide an overview of the relevant literature regarding the estimation of the different dimensions of user search queries since different studies have been proposed to solve this problem. We specifically target works dealing with implicit time-sensitive queries, which in contrast to recency-sensitive ones (Efron and Golovchinsky 2011; Li and Croft 2003), approach results that are preferably from a specific time period. The methods proposed to solve this problem can be broadly classified into three different classes: (1) metadata-based, (2) query log-based and (3) content-based approaches.

Within the overall context of T-IR, Jones and Diaz (2007) were the first to consider implicit time sensitive queries. In their work, the authors follow a metadata-based approach by using a language model trained over a collection of web news documents to model the period of time that is relevant to a query. More specifically, they estimate distribution $P(t|q)$, where t is the day relevant to query q. They adopt a relevance modeling solution that considers, not only the probability of the document's relevance, given by $P(q|d)$, but also the temporal information about the document, given by $P(t|d)$, where t is the day relevant to that document (note that this probability equals *0* if *t* day is not equal to the document timestamp and *1* otherwise). Kanhabua and Nørvåg (2010) proposed three different methods to determine the time of implicit time sensitive queries: (1) dating queries using only query keywords, (2) dating queries using the retrieved top-*k* documents, and (3) dating queries using the timestamp of the retrieved top-*k* documents. They rely on the use of temporal language models, based on a New York Times (*NYT*) news collection, where documents are explicitly time-stamped with the document creation time. Dakka et al. (2012) proposed a solution which takes into account the publication times of documents to identify the important time intervals that are likely to be of interest to an implicit temporal query. Time is incorporated into language models to assign an estimated relevance value to each time period. Alongside this, other works have looked into improving the retrieval effectiveness of implicit temporal queries. Peetz et al. (2014) for example, proposes to leverage temporal bursts of documents (based on document timestamp) to develop a query modeling approach that incorporates a selection of the most descriptive terms of the documents. Unfortunately, all of these approaches rely on the creation date of documents as correct temporal signals, which are simply not available in many documents.

An alternative solution to using metadata was proposed by Vlachos et al. (2004) who developed a method to discover valuable time periods using the query logs of a commercial search engine. Likewise, Metzler et al. (2009) suggested mining query logs to identify implicit temporal information needs. They proposed a weighted measure that considers the number of times a query is pre- and post-qualified with a given year (e.g., "*Miss Universe 1990*" and "*Miss Universe 1991*"). A relevance value is then given for each year found in a document. Based on this, they proposed a time-dependent ranking model that explicitly adjusts the score of a document in favor of those matching the users' implicit temporal intents. The referred study addresses an interesting solution because it introduces the notion of correlation between a query and a year. However, the approach lacks query coverage since it depends on the analysis of query logs, which are not easily available. Another research is the work of Shokouhi and Radinsky (2012) who proposed a time-sensitive approach for query auto-completion by applying time series analysis. Their results show that predicting the popularity of queries by time series analysis and periodicity estimation is more reliable than straightforwardly using information on past query popularity derived from web query logs.

While the above models rely on spikes in the distribution of relevant documents or queries, none extracted temporal information from web contents in order to date implicit time sensitive queries. The closest prior research to our work was proposed by Gupta and Berberich (2014), Strötgen et al. (2012) and by Foley and Allan (2015). More specifically, Gupta and Berberich (2014) propose an interesting initial work that makes use of both metadata (document's timestamp), as well as content information (temporal expressions from their contents), to identify times of interest to a given query. However, instead of considering single points in time (e.g., the years 1983, 1990, 2000, 2011), they follow a parallel line of research to ours that focus on determining time intervals of interest to the query. For example, for the query "*amy winehouse*", this would stem in determining the period ([1983, 2011]) which covers her lifetime instead of retrieving multiple single time references (e.g., 1983; 2011) or other related times occurring during this period, for instance 2003 (her first appearance in stage) or 2006 (back to black album). In clear contrast to this work, Strötgen et al. (2012) set forth the first approach to identify the most relevant temporal expressions with information extracted from text documents. Each temporal expression is represented by a set of document and corpus-based features. The relevance of the temporal expressions is combined into a single relevance function based on a set of pre-defined heuristics. Although an interesting approach, this works aims to identify relevant temporal expressions within documents themselves, rather than that of determining the time of a time-sensitive query. Foley and Allan (2015) in turn, address the problem of selecting relevant years to queries as an unsupervised re-ranking problem. In particular, they rely on the query likelihood model and the sequential dependence model, to model the similarity between queries and documents, under a language model framework that is built from all the sentences mentioning a particular year. Years are then ranked according to a reciprocal rank weighting, which assigns every occurrence of a year a score equal to 1/rank.

In addition to the above-mentioned studies, two other proposals, Kawai et al. (2010) and Kanhabua et al. (2012), have been implemented to tackle the problem of filtering noisy temporal expressions. The method put forward by Kawai et al. (2010) suggested an approach to filter out noisy year expressions from web snippets that are temporally irrelevant to the query by applying machine learning techniques trained over a set of labeled triplets. Each triplet consists of a sentence, a query and the temporal expression found in the sentence. Although the incorporation of a date filtering process is novel, their proposal does not determine the degree of relevance for each temporal pattern. Similarly to the above method, Kanhabua et al. (2012) propose to identify relevant temporal expressions but this time with regard to a particular event, specifically a place or a named entity relevant to the medical domain.

Our approach differs from previous research on dating queries in several aspects. Firstly, we consider single time points instead of time intervals. Secondly, instead of making use of query logs or metadata information we rely on the documents' contents. Moreover, we do not resort to a set of heuristics extracted from a document's content or a supervised classification methodology. Instead, in our approach, we detect relevant temporal expressions based on corpus statistics and a general similarity measure that makes use of co-occurrences of words and years extracted from the contents of the web documents. This means that we could easily have a date deemed as relevant while only appearing in a single document, as the inverse. Finally, apart from estimating the degree of relevance of a temporal expression, we present, in addition, an appropriate classification strategy to determine whether or not a date is relevant to the query.

4 Identifying query relevant temporal expressions

In this section, we describe the method that guides our identification of top relevant dates related to text queries with a temporal dimension. To tackle this problem, we adopt a web content analysis approach that extracts temporal information from the top-n web results returned in response to a query. The overall idea of the process is to identify and classify calendar years that are relevant for a given query on four different steps depicted in Fig. 2 and explained in the remainder of this section: web search, text representation, temporal similarity and date filtering.

4.1 Web search

We assume a query to be either explicit, i.e., a combination of both text and time, denoted q_{time}, or implicit, i.e., just text, denoted q_{text}. In this article, we deal with the latter since handling explicit temporal queries is a less complex task. For the sake of readability, we

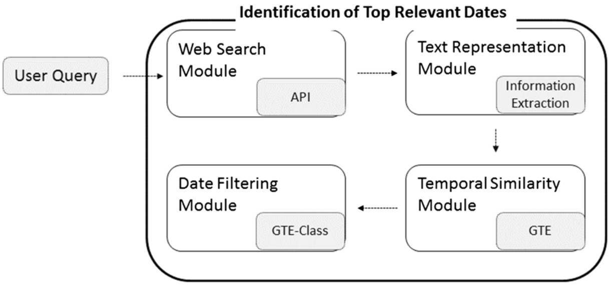

Identification of Top Relevant Dates
User Query
Web Search Module
API
Text Representation Module
Information Extraction
Date Filtering Module
GTE-Class
Temporal Similarity Module
GTE

Fig. 2 Overall architecture

denote a query simply as q. Similarly to Kawai et al. (2010), we use a prospective search where the query is first issued before results are gathered and indexed. For the purposes of collecting the results, we use a web search API to access an up-to-date index search engine. Given a text query q, we obtain, as the result of the search, a collection of n web text results $T = \{T_1, T_2, \ldots, T_n\}$.

4.2 Text representation

Each T_i, for $i = 1, \ldots, n$, denotes the concatenation of two texts, i.e., $\{Title_i, Text_i\}$ and is represented by a bag-of-relevant-words and a set of candidate temporal expressions. In what follows, we assume that each T_i is composed by two different sets denoted W_{T_i} and D_{T_i}:

$$T_i \rightarrow (W_{T_i}, D_{T_i}), \tag{1}$$

where $W_{T_i} = \{w_{1,i}, w_{2,i}, \ldots, w_{k,i}\}$ is the set of the k most relevant words/multiwords associated with a text T_i and $D_{T_i} = \{d_{1,i}, d_{2,i}, \ldots, d_{t,i}\}$ is the set of the t candidate years associated with a text T_i. Moreover,

$$W_T = \bigcup_{i=1}^{n} W_{T_i}, \tag{2}$$

is the set of distinct relevant words/multiwords (hereafter called words) extracted for a query q, within the set of texts T, i.e., the relevant vocabulary. In this article, relevant words are identified for any text based on a specific segmentation process and a numeric selection heuristic. Similarly,

$$D_T = \bigcup_{i=1}^{n} D_{T_i}, \tag{3}$$

is defined as the set of distinct candidate years extracted from the set of all texts T. For the recognition of dates a simple rule-based model was applied to extract the following explicit temporal patterns: *YYYY*, *YYYY-YYYY*, *YYYY/YYYY*, *MM/dd/YYYY*, *dd/MM/YYYY*, *MM.dd.YYYY* and *dd.MM/YYYY*. Further alternatives were to use HeidelTime[1] temporal

[1] http://dbs.ifi.uni-heidelberg.de/index.php?id=form-downloads [December 22, 2016].

tagger (Strötgen and Gertz 2015) which is better suited to more complex tasks than that of extracting explicit temporal patterns. Each discovered pattern is then normalized to the *YYYY* granularity level. Note, however, that a document can also contain other types of temporal expressions, other than explicit ones. This includes implicit and relative temporal expressions. Nevertheless, these ones will not be studied in this article as they require linguistic pre-processing steps that lie outside the scope of this work. Finally, W_j^* is defined as the set of relevant words W_T that appear together with the candidate date d_j in any text T_i.

To illustrate our approach, we present a running example for the query "*Haiti earthquake*". Table 1 lists the set of three texts retrieved upon the query execution and the formed sets, W_{T_i} and D_{T_i}.

Let $W_T = \{$*haiti earthquake*; *major earthquakes*; *haiti*; *catastrophic damage*; *Port-au-Prince*; *Concepción de la Vega*$\}$ be the set of distinct relevant words, $D_T = \{$*1500*; *1564*; *2010*; *2011*$\}$ the set of candidate dates and W_j^* as the set of relevant words W_T that co-occur with each of the four candidate dates D_T in any text (see Table 2).

Each candidate date is then assessed with regard to its temporal similarity with the query. We formalize this process in the following section.

4.3 GTE: temporal similarity measure

In this section, we introduce our temporal similarity measure which evaluates the temporal relatedness between a query and a candidate date. We formally define this problem as follows: given a query q and a candidate date $d_j \in D_T$ assign a degree of relevance to each (q, d_j) pair. To model this relevance, we will use a temporal similarity measure, *SIM*, to be defined, ranging between 0 and 1:

$$SIM(q, d_j) \in [0, 1]. \tag{4}$$

Table 1 Running example: *Haiti earthquake*

Title$_1$	**2011 Haiti Earthquake** Anniversary
Text$_1$	As of **2010** (see **1500** photos here), the following **major earthquakes** have been recorded in **Haiti**. The first one occurred in **1564**.
W_{T_1}	*haiti earthquake; major earthquakes; Haiti*
D_{T_1}	*1500; 1564; 2010; 2011*
Title$_2$	**Haiti Earthquake** Relief
Text$_2$	On January 12, **2010**, a massive earthquake struck the nation of **Haiti**, causing **catastrophic damage** inside and around the capital city of **Port-au-Prince**.
W_{T_2}	*haiti earthquake; haiti; catastrophic damage; Port-au-Prince*
D_{T_2}	*2010*
Title$_3$	**Haiti Earthquake**
Text$_3$	The first great earthquake mentioned in histories of **Haiti** occurred in **1564** in what was still the Spanish colony. It destroyed **Concepción de la Vega**.
W_{T_3}	*haiti earthquake; haiti; Concepción de la Vega*
D_{T_3}	*1564*

Words in bold correspond to Wt; Numbers in bold to Dt

Table 2 List of words W_T that co-occur with the candidate dates

W_T	W^*_{1500}	W^*_{1564}	W^*_{2010}	W^*_{2011}
Haiti earthquake	X	X	X	X
major earthquakes	X		X	X
Haiti	X	X	X	X
catastrophic damage			X	
Port-au-Prince			X	
Concepción de la Vega		X		

In each column the "X" indicate the words belonging to W^*_j

The aim is to identify dates d_j, which are relevant for q and minimize any errors caused by non-relevant or wrong dates.[2] Our proposal is that the relevance between a (q, d_j) pair is better defined if, instead of just focusing on the self-similarity between the query q and the candidate date d_j, all the information existing between W^*_j and d_j is considered. Considering the candidate date *2010* in our running example, this means that we should consider not only the similarity between *2010* and the query "*Haiti earthquake*", but also all the similarities occurring between *2010* and W^*_{2010}, identified in Table 2 with an "X".

Similarly, we should process all the similarities between *1500*, *1564*, *2011* and the corresponding W^*_j. Our assumption is based on the following principle:

P1: The more a given candidate date is correlated to the set of corresponding, distinct and most relevant words associated with the query—i.e., the intersection between the set of words relevant with the query, W_T, and the set of words W^*_j co-occurring with the candidate date d_j—the more the query will be associated with the candidate date.

Thus, we will not only define the similarity between the query words q and the candidate date d_j, but also between each of the most important words $w_{\ell,j} \in W^*_j$ and the respective candidate date d_j. Our proposal for the measure *SIM* is GTE (Generic Temporal Value), which is presented in Eq. 5, where *sim* represents any similarity measure of first or second order and F an aggregation function (Max/Min; Arithmetic Mean; Median) of the several $sim(w_{\ell,j}, d_j)$:

$$GTE(q, d_j) = F(sim(w_{\ell,j}, d_j)), w_{\ell,j} \in W^*_j. \tag{5}$$

We describe each of these two topics, *sim* and F, as follows.

4.3.1 Similarity measure

In this article, *sim* represents a similarity measure, either of first or second order. While first order association measures (e.g., DICE) evaluate the relatedness between two tokens as they co-occur in a given context (e.g., ngram, sentence, paragraph, corpus), second order measures are based on the principle that two words are similar if their corresponding context vectors are also similar thus following Harris distributional hypothesis (Harris 1954). The intuition behind second order similarity measures is that two terms having many co-occurring words often carry the same sense in such a way that the information content of both words is likely to share similar terms. For instance, the similarity between

[2] We understand non-relevant dates as temporal patterns which though being dates are not relevant to the query (e.g., avatar movie 2011) and wrong ones as those, which though being a temporal pattern do not form a data (e.g., 1500 photos).

the terms "*professor*" and "*teacher*" is expected to rest on a number of common co-occurring words such as student, school, etc. Adopting one such solution, will enable to overcome the problem of data sparseness in cases when two terms, despite being similar, do not co-occur frequently in a corpus in order to model language accurately.

Figure 3 shows an example for both types of measures. In the figure, d_j represents one candidate date, for instance *2010* and $w_{\ell,j}$ represents one of the several possible words of W^*_{2010}, for example *Port-au-Prince*. X and Y in turn represent the context vectors of $w_{\ell,j}$ and d_j, a set of tokens that co-occur somehow with the target word and the target candidate date respectively. The rationale is that the co-occurrence between *Port-au-Prince* and *2010* is not enough to convey their similarity as done by their corresponding context vectors which are expected to contain similar related terms co-occurring with them within a certain predefined window.

Our hypothesis, which will be supported in the experiments section, is that second order similarity measures carry valuable additional relations in both the word $w_{\ell,j}$ and the candidate date d_j context vectors, which cannot be induced if a direct co-occurrence approach between $w_{\ell,j}$ and d_j is used.

In this work, we apply the InfoSimba (*IS*) second-order similarity measure, a measure supported by corpus-based token correlations proposed by Dias et al. (2007) as defined in Eq. 6:

$$IS\left(w_{\ell,j}, d_j\right) = \frac{\sum_{i\in X}\sum_{j\in Y} S(i,j)}{\left(\begin{array}{c}\sum_{i\in X}\sum_{j\in X} S(i,j)+\\ \sum_{i\in Y}\sum_{j\in Y} S(i,j)-\\ \sum_{i\in X}\sum_{j\in Y} S(i,j)\end{array}\right)}. \tag{6}$$

IS calculates the correlation between all pairs of two context vectors X and Y, where X is the context vector representation of $w_{\ell,j}$, Y is the context vector representation of d_j and $S(.,.)$ is any symmetric similarity measure. To define the context vectors, we have at least five possible representations: (W;W), (D;D), (W;D), (D;W) and (WD;WD), where W stands for a word-only context vector, D for a date-only one and WD for a combination of words and dates. A clear picture of all the possible representations is given in Fig. 4, where $(w_1, w_2, .., w_k)$ and $(d_1, d_2, .., d_t)$ are the elements of the two context vectors, i.e., the set of elements of W_T and D_T, respectively. The best possible representation will be determined later on the experimental section.

Furthermore, we have to define the size of the context vector, denoted N and a threshold similarity value TH. This threshold is the minimum similarity value above which, words and candidate dates should be selected as elements of the two context vectors. For instance, to determine the context vector of a candidate date d_j for the representation type (WD;WD), only those words $(w_1, w_2, .., w_k)$ and candidate dates $(d_1, d_2, .., d_t)$ having a minimum S similarity value ($S > TH$) with $\left(., d_j\right)$[3] are eligible for the N-size context vector, where S is any first order similarity measure (e.g., Pointwise Mutual Information, Symmetric Conditional Probability or Dice coefficient). Likewise, S would relate all the possible combinations $(w_j, .)$ that would enable us to determine the set of words $(w_1, w_2, .., w_k)$ and candidate dates $(d_1, d_2, .., d_t)$ that should be part of the w_j N-size context vector.

We illustrate this in Table 3 showing the M_{ct} matrix from our running example, a conceptual temporal correlation matrix, which stores the S similarity between the most

[3] i.e. that co-occur at least once with d_j.

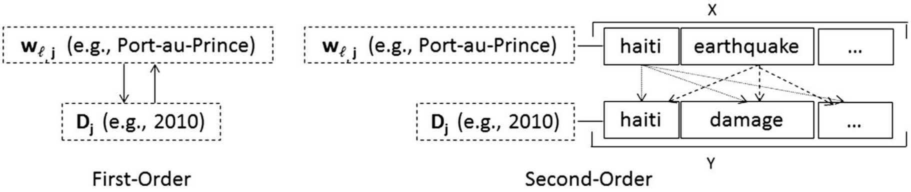


Fig. 3 Example of first order and second order similarity measures

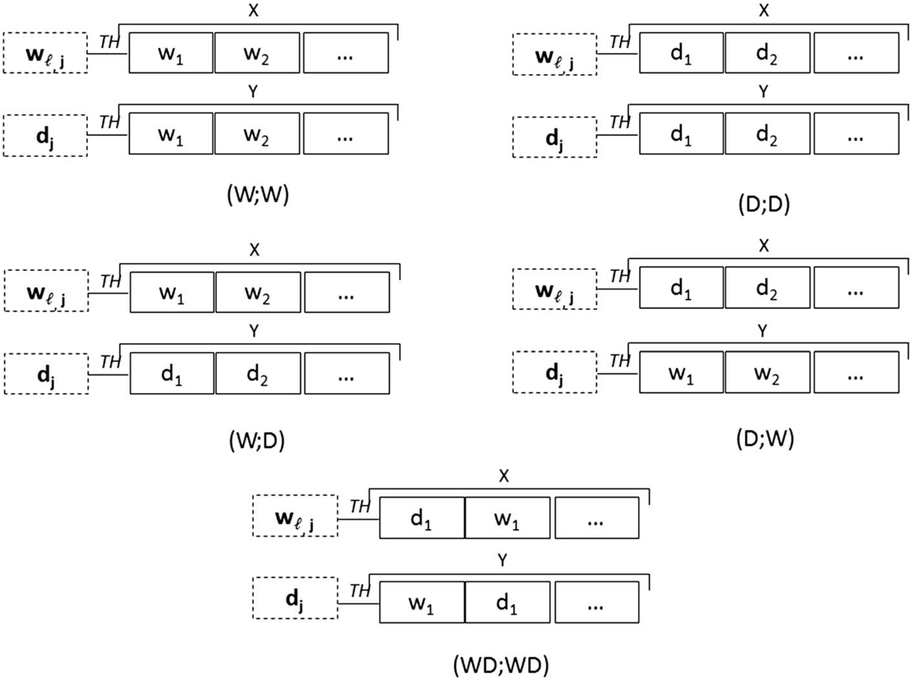


Fig. 4 Context vector representations: (***W***;***W***), (***D***;***D***), (***W***;***D***), (***D***;***W***), (***WD***;***WD***)

important words and the candidate dates. We focus on calculating the DICE coefficient[4] for the candidate date *2010* and for the relevant word *Port-au-Prince*. Based on the above representation and on a threshold $TH > 0$ we determine the eligible context vectors for both *2010* and *Port-au-Prince*. The result is a list whose components are arranged in the descending order of the similarity value. As such, we obtain (*Haiti earthquake*, *Haiti*, *major earthquakes*, *catastrophic damage*, *Port-au-Prince*, *1500*, *2011*, *1564*) for *2010* and (*catastrophic damage*, *2010*, *Haiti earthquake*, *Haiti*) for *Port-au-Prince*. After defining N we may then determine the final version of the context vectors. For example, if N is set to 2, we will have (*Haiti earthquake*, *Haiti*) as the context vector of *2010* and (*catastrophic damage*, 2010) as the final context vector of *Port-au-Prince*.

IS can now be computed as the corresponding similarity between each pairs of tokens (words *or*/*and* dates), present in the N-size context vectors as depicted in Fig. 5.

[4] Please refer to Eq. 15 in case you need to recall DICE coefficient.

Table 3 M_{ct} matrix for our running example

	Haiti earthquake	major earthquakes	Haiti	catastrophic damage	Port-au-Prince	Concepción de la Vega	1500	1564	2010	2011
Haiti earthquake	1	0.5	1	0.5	0.5	0.5	0.5	0.8	**0.8**	0.5
major earthquakes	0.5	1	0.5	0	0	0	1	0.6	**0.66**	1
Haiti	1	0.5	1	0.5	0.5	0.5	0.5	0.8	**0.8**	0.5
catastrophic damage	0.5	0	0.5	1	1	0	0	0	**0.66**	0
Port-au-Prince	**0.5**	0	**0.5**	**1**	1	0	0	0	**0.66**	0
Concepción de la Vega	0.5	0	0.5	0	0	1	0	0.66	0	0
1500	0.5	0.5	0.5	0	0	0	1	0.66	**0.66**	1
1564	0.8	0.66	0.8	0	0	0.66	0.66	1	**0.5**	0.66
2010	0.8	0.66	0.8	0.66	0.66	0	0.66	0.5	1	0.66
2011	0.5	1	0.5	0	0	0	1	0.66	**0.66**	11

Numbers in bold means a S similarity value with a threshold TH > 0

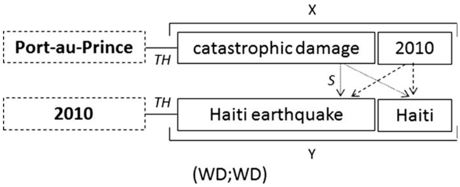


Fig. 5 (***WD***;***WD***) context vector representation for *Port-au-Prince* and *2010*

Specifically, it will compute the level of relatedness between *catastrophic damage* from the context vector of *Port-au-Prince* and the two other context tokens of *2010*—i.e., *Haiti earthquake*, *Haiti*—and then between *2010* and all other context tokens of *2010* and so on, thus promoting semantic similarity. Note that the similarity between each pair of tokens is again determined by *S*, which in our example is the Dice coefficient measure. We recall that this measure was already used to determine the set of best tokens that should be part of the context vectors. The final score of $IS(Port - au - Prince, 2010)$ which stems from applying Eq. 6 is given by:

$$\frac{\begin{array}{c} S(catastrophic\, damage, haiti\, earthquake) + S(catastrophic\, damage, haiti) \\ +S(2010, haiti\, earthquake) + S(2010, haiti) \end{array}}{\begin{array}{c} (S(catastrophic\, damage, catastrophic\, damage) + S(catastrophic\, damage, 2010) \\ +S(2010, catastrophic\, damage) + S(2010, 2010)) \\ + \\ \begin{pmatrix} S(haiti\, earthquake, haiti\, earthquake) + S(haiti\, earthquake, haiti) \\ +S(haiti, haiti\, earthquake) + S(haiti, haiti) \end{pmatrix} \\ (S(catastrophic\, damage, haiti\, earthquake) + S(catastrophic\, damage, haiti) \\ +S(2010, haiti\, earthquake) + S(2010, haiti)) \end{array}}$$

By looking at the similarities stored on the M_{ct} matrix we can then compute the final value as follows:

$$IS(Port - au - Prince, 2010) = \frac{0.5 + 0.5 + 0.8 + 0.8}{\begin{array}{c} (1 + 0.66 + 0.66 + 1)+ \\ (1 + 1 + 1 + 1)- \\ (0.5 + 0.5 + 0.8 + 0.8) \end{array}} = \frac{2.6}{4.72} = 0.55$$

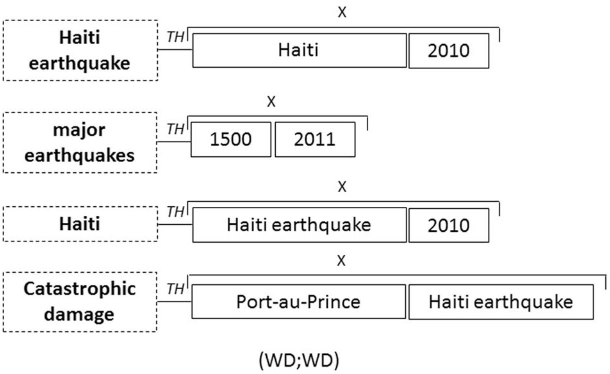


Fig. 6 (***WD***;***WD***) context vector representation for *Haiti earthquake, major earthquakes, Haiti* and *catastrophic damage*

Similarly, we should process all the IS similarities between *2010* and the remaining words of W^*_{2010}, i.e., *catastrophic damage*, *Haiti*, *major earthquakes* and *Haiti earthquakes*. Figure 6 shows the *X* context vectors of all these words. The *Y* context vector of 2010 has already been introduced in Fig. 5.

The final score of each computation is given as follows:

$$\begin{aligned} &IS(\text{catastrophic damage}, 2010) = 0.7;\\ &IS(\text{haiti}, 2010) = 0.9;\\ &IS(\text{major earthquakes}, 2010) = 0.33;\\ &IS(\text{haiti earthquake}, 2010) = 0.9. \end{aligned}$$

Next, we describe the *F* aggregation function which is used to combine the several smilarity values $sim\left(w_{\ell,j}, d_j\right)$, computed by IS.

4.3.2 Aggregation function

In order to combine the different similarity values produced for the candidate date, d_j, in a single value capable of representing its relevance to the query words, we propose an aggregation function *F*. With that objective in mind, we consider three different *F* functions:

1. The Max/Min;
2. The Arithmetic Mean;
3. The Median.

While the Mean and the Median are measures of central tendency, the Max/Min approach relies on extreme values. To understand this approach more adequately, we establish two requirements for MAX and MIN values.

R1 (MAX): the higher the number of relevant words related to the candidate date, the higher the similarity. To enter the specifics, the system selects the maximum similarity within all the $\left(w_{\ell,j}, d_j\right)$ similarity values if the proportion of relevant words which appear with the candidate date is above a given threshold θ. In this case, θ has experimentally been defined as *0.2*.

R2 (MIN): the lower the number of relevant words related to the candidate date, the lower the similarity. As such, proportion values ≤ 0.2 result in simply selecting the $sim\left(q, d_j\right)$ as a similarity value. This is often the minimum one.

As we shall see in the experiments section, best results occur for the median aggregation function. If we apply this function to the IS similarity values obtained from our running example (Median(0.55, 0.7, 0.9, 0.33, 0.9)) then we will get a final score of 0.7. Instead, a score of 0.66 would have been reached if a first order similarity measure such as DICE would have been applied to the same set of tokens (Median(0.66, 0.66, 0.8, 0.66, 0.8)). Though anecdotally, this example shows that one such second order similarity measure, such as IS, is able to produce better results than a first order similarity measure as a score of 0.7 better reflects the high similarity that exists between *Port-au-Prince* and *2010.* An overall analysis of the experimental results will later on confirm these introductory examples on Sect. 5.2.1.

4.3.3 Overall procedure

The overall strategy of our query tagging relevance model is shown in Algorithm 1.

Algorithm 1: Assign a degree of relevance to each (q, d_j) pair

```
Input: query $q$
T ← GetResultsFromSearchEngine($q$)
For each $T_i \in T$, i = 1,..,n
    Apply Text Processing
    $W_{T_i}$ ← Select best relevant words/multiwords in $T_i$
    $D_{T_i}$ ← Select all temporal patterns in $T_i$
$W_T \leftarrow \bigcup_{i=1}^{n} W_{T_i}$
$D_T \leftarrow \bigcup_{i=1}^{n} D_{T_i}$
Compute $M_{ct}$
For each $d_j \in D_T$
    Compute $GTE(q, d_j)$
Output: $V_{GTE_{D_T}}$ relevance
```

The algorithm receives a query from the user, fetches related web results from a given search engine and applies text processing to the set of texts. This processing task involves selecting the most relevant words and collecting the candidate years in each web result. The words and candidate years' matrix is then computed. Finally, each candidate year is given a temporal similarity value to the query computed by $GTE(q, d_j)$. The final relevance results are kept in a new vector called $V_{GTE_{D_T}}$ defined in Eq. 7:

$$V_{GTE_{D_T}} = \begin{bmatrix} TS_1 \\ TS_2 \\ \vdots \\ TS_t \end{bmatrix}, \tag{7}$$

where $TS_k, k = 1, \ldots, t$ represents the temporal similarity between a candidate date d_j, and the query q, for the t distinct candidate dates. In the following section, we describe the last step of our approach.

4.4 GTE-class: date filtering

Our next step is to define an appropriate classification strategy to determine whether the candidate temporal expressions are actually relevant or not. One such approach, which we designate as *GTE-Class,* will enable any system to leverage this information in order to improve the effectiveness of the results presented to the user. We could use this for example to filter out non-relevant dates from the output of a temporal clustering solution, or to improve the ranking of the results by considering a value of 0 as opposed to the similarity value determined, as low as it is. To accomplish this objective, we suggest a classical threshold-based strategy. Thus, given a (q, d_j) pair, the system will automatically classify a date based on the following expression:

1. Relevant, if $GTE(q, d_j) \geq \lambda$,
2. Non-relevant or wrong date, if $GTE(q, d_j) < \lambda$,

where λ has to be tuned to at least a local optimum.

An illustration of this is given in Eq. 12 for $\lambda = 0.35$. A more thorough discussion of this value, along with many more experiments, can be found on Sect. 5.2.2. The final set of m relevant dates for the query q is defined by D_T^{Rel}:

$$D_T^{Rel} = \{d_1^{Rel}, d_2^{Rel}, \ldots, d_m^{Rel}\}, \tag{8}$$

where $d_1^{Rel} < d_2^{Rel} < \cdots < d_m^{Rel}$. Note that $d_1^{Rel} and d_m^{Rel}$ represent the lower and the upper temporal bounds of the query q respectively. Similarly, D_{T_i} is defined as:

$$D_{T_i}^{Rel} = \left\{d_{1,i}^{Rel}, d_{2,i}^{Rel}, \ldots, d_{u,i}^{Rel}\right\}, \tag{9}$$

where u represents the set of of u relevant dates $d_{j,i}, j = 1, ..u$ for the query q associated with the text T_i. Based on this, each text T_i is no longer represented by a set of candidate temporal expressions, but by a set of relevant dates. We redefine T_i as follows:

$$T_i \to \left(W_{T_i}, D_{T_i}^{Rel}\right). \tag{10}$$

Finally, $V_{GTE_{D_T}}$ becomes $V_{GTE_{D_T}^{Rel}}$ such that:

$$V_{GTE_{D_T}^{Rel}} = \begin{bmatrix} GTE_1 \\ GTE_2 \\ \vdots \\ GTE_m \end{bmatrix}, \tag{11}$$

where $GTE_k, k = 1, \ldots, m$ represents the temporal similarity between the date d_j and the query q, for the m distinct relevant dates and $m \leq t$. This is illustrated as follows:

$$V_{GTE_{D_T}} = \begin{array}{c} d_1 \\ d_2 \\ d_3 \\ d_t \end{array} \begin{bmatrix} 0.2 \\ 0.6 \\ 0.3 \\ 0.8 \end{bmatrix} \xrightarrow[\text{GTE - Class}]{} V_{GTE_{D_T}^{Rel}} = \begin{array}{c} d_1 \\ d_2 \\ d_3 \\ d_t \end{array} \begin{bmatrix} - \\ 0.6 \\ - \\ 0.8 \end{bmatrix} = \begin{array}{c} d_1 \\ d_2 \end{array} \begin{bmatrix} 0.6 \\ 0.8 \end{bmatrix}. \tag{12}$$

Note that the candidate date d_1 and d_3 are both filtered out from the final list $V_{GTE_{D_T}^{Rel}}$, as they have been classified by GTE-Class as a non-relevant temporal pattern either by the threshold-based strategy or the supervised learning process. In the following section, we define the experimental setup.

5 Experiments

In this section, we describe the experimental results of our work. In order to evaluate the effectiveness of our system we perform a number of experiments to test the GTE similarity measure and the GTE-Class date filtering process under two different collections. The first one based on web snippets, and the second one based on the full text of web pages, each with its own set of queries. This will give us some insight into how much, if at all, our system behaves better under a small collection of texts, or if, for example, the technique performs poorly under a different set of queries. The rest of this section is organized as follows. Section 5.1 describes the set of queries and the collection used in our experiments. Section 5.2 discusses the results obtained.

5.1 Queries and collection

Evaluating time-sensitive information needs is a difficult task since there are no available benchmarks such as TREC[5] bringing together time sensitive queries and associated temporal tags. Over the years a few reference collections have been set but they often consist of newswire articles with temporal information being inferred from the timestamp of the document rather than from the document's contents. The TREC 2004 Novelty track,[6] for example, created a set of 50 queries of which only a few are explicitly tied to a single dated event though they can have multiple temporal instances associated. The system is designed to locate relevant and new information within a set of newswire documents, thus query judgments are not taken into account. Another source of TREC queries is based on the TREC 2004 Robust Track[7] news corpus, which gathers some time-sensitive ad hoc queries selected from TREC-{6,7,8} and previous robust tracks. Likewise, the novelty dataset, this collection is not designed to determine the correct time of the query. Recently two other temporal tasks have been launched. The TREC 2013 and 2014 Temporal Summarization[8] (Guo et al. 2013) task consists of 25 temporal queries and a set of timestamped documents covering the time period October 2011 through April 2013. Its goal is geared towards the summarization of events over time. This contrasts with our approach, which aims to infer from the web contents and not from the timestamps of the documents the different possible times of a query. Finally, the NTCIR-11-Temporalia[9] (Joho et al. 2014) challenge comprises a document corpus of blog and news sources and a mixed combination of implicit and explicit temporal queries, which is far away from our purpose.

Given the absence of a TREC or similar collection that suits our temporal information retrieval task, we developed two new publicly available datasets (*WC_DS*[10] and *WC_TREC_DS*[11]), gathering implicit temporal queries, documents, and date relevance judgments, thus establishing baseline performance for further studies. We note that collections vary both by type (WC_DS is a web snippets collection, while WC_TREC_DS is a web full text collection), number of queries, documents and date candidates, thus providing a diverse experimental setup for assessing the robustness of our similarity measure and classification procedure. A summary of the collections used in our experiments is given in Table 4. A detailed explanation of each one is given in Sects. 5.1.1 and 5.1.2 respectively for WC_DS and WC_TREC_DS.

Note that because of the lack of a public collection that provides temporal relevance for time-sensitive queries (i.e., which of the dates of a set of texts are relevant with a query) it becomes hard to conduct experiments with a larger number of queries. The number of queries used in our experiments however is in line with related research (Jatowt and Yeung 2011; Jones and Diaz 2007; Kanhabua and Nørvåg 2008; Kanhabua et al. 2011). A step forward in our work is that in addition to the experiments carried out we provide a demo search interface[12] (Campos et al. 2014b), thus catering users with the possibility of

[5] http://trec.nist.gov [December 22, 2016].

[6] http://trec.nist.gov/data/t13_novelty.html [December 22, 2016].

[7] http://trec.nist.gov/data/t13_robust.html [December 22, 2016].

[8] http://www.trec-ts.org [December 22, 2016].

[9] http://ntcir.nii.ac.jp/Temporalia/NTCIR-11-Temporalia/ [December 22, 2016].

[10] http://www.ccc.ipt.pt/~ricardo/datasets/WC_DS.html [December 22, 2016].

[11] http://www.ccc.ipt.pt/~ricardo/datasets/WC_TREC_DS.html [December 22, 2016].

[12] http://tm-websuiteapps.ipt.pt/GTEAspNetFlatTempCluster_Server/ [December 22, 2016].

Table 4 Summary of WC_DS and WC_TREC_DS collections

Name	#Queries	#Docs	#Docs with Dates	# (q, d_j) pairs
WC_DS	42	2100	582	235
WC_TREC_DS	25	1250	489	443

extensively testing the effectiveness of our proposal by running a set of queries against our algorithm. In response to a query submitted in a search box, our algorithm displays a set of dates generated "on the fly". Each date is assigned a temporal similarity value reflecting its similarity with the user query, i.e., its BGTE. We believe this is a valuable contribution to the research community.

5.1.1 WC_DS collection

For the WC_DS collection, we rely on Google Trends—a Google service that provides users with a visual representation of top and rising searches—to gather a representative set of queries. We start by selecting 20 queries per each of the 27 pre-defined available categories. After removing duplicates and explicit temporal queries, we end up with a set of 450 queries. As we aim to evaluate the temporal similarity between a query and a set of candidate dates, we need to guarantee that the queries selected are non-ambiguous in concept and temporal in their purpose, such that each query is well-defined in terms of relevant dates. For the first step, we used the Wikipedia disambiguation feature, which helps to understand whether a query has more than one meaning or facet. Final results show that 176 queries are of clear nature, i.e., non-ambiguous. Each clear concept query must then be classified with regard to its temporal nature. Following the work of Jones and Diaz (2007) we define two temporal classes: (1) *Atemporal*, i.e., queries not sensitive to time (e.g., "*rabbit*"); (2) *Temporal*, i.e., queries that either take place in a very concrete time period, known as temporally unambiguous (e.g., "*bp oil spill*") or that have multiple instances over time, known as temporally ambiguous (either occurring in a *periodic* fashion—e.g., "*SIGIR*"—or in an uncertain aperiodical manner—e.g., "*oil spill*"). For the purpose of judging the set of 176 clear concept queries with regard to their temporality, three human annotators were asked to consider each query, to look at web search results and to classify them as Temporal or Atemporal. As an alternative to this manual identification, we could have resorted to some temporal categorization strategy, either Wikipedia or snippet-based (Campos et al. 2011a). We opt not to use any of these approaches as our intention was to stick as close as possible to the real ground truth, i.e., people, without introducing any potential error into the classification scheme.

The final classification of each query comes by majority voting. As such, each query is considered to be Atemporal if it gets at least two votes, while Temporal otherwise. An inter-rater reliability analysis using the Fleiss Kappa statistics (1971) was then performed to determine consistency among annotators. Results have shown a value of 0.89, thus indicating an almost perfect agreement between the raters. The final set (see Table 5) consists of 42 real-world text clear-concept temporal queries.

Based on the 42 text queries, we developed a web content dataset (*WC_DS*) consisting of 235 (q, d_j) pairs. For this, we queried the Bing search engine collecting the top-50 web

Table 5 List of text queries

george bush iraq war	avatar movie	tour eiffel	steve jobs	amy winehouse
slumdog millionaire	britney spears	troy davis	waka waka	haiti earthquake
football world cup	justin bieber	adele	nissan juke	marco simoncelli
walt disney company	little fockers	swine flu	dan wheldon	volcano iceland
lena meyer-landrut	kate middleton	ryan dunn	david villa	true grit
california king bed	bp oil spill	fiat 500	Haiti	susan boyle
sherlock holmes	tour de france	lady gaga	katy perry	dacia duster
fernando alonso	david beckham	Fukushima	Obama	kate nash
osama bin laden	rebecca black			

snippets, using for this purpose the Bing Search API,[13] parameterized with the *en-US* market language parameter. We argue that snippets, as shown by Alonso et al. (2009a), are an interesting alternative collection for the representation of web documents, which provides a short summary of the document where dates, in the form of years often appear. Of the total number of 2100 web snippets retrieved, only those annotated with at least one candidate year term were selected. The final set consists of 582 web snippets S_i with years and 235 distinct $\left(q, d_j\right)$, where q is the query and d_j the candidate year. The ground truth was then obtained by automatically labeling each one of the 235 distinct $\left(q, d_j\right)$ pairs. In order to do this, we followed a twofold approach:

1. Each $(S_i, d_{j,i})$ is manually assigned a relevance label on a 2-level scale: not a date or temporally non-relevant to the query within a snippet S_i (score 0) and temporal relevant to the query within a snippet S_i (score 1). The labeler was allowed to perform a search on the web, so as to produce knowledge about the topic and eliminate context factors that might influence a change in his judgment. The final list of judgments consists of 119 $(S_i, d_{j,i})$ labeled with score 0, and 537 with score 1.
2. Each $\left(q, d_j\right)$ pair is then automatically labeled based on Eq. 13:

$$\left(q, d_j\right) = \begin{cases} 1, if\#\#\#\mathrm{Re}l \geq \#\#\overline{\mathrm{Re}l} \\ 0, if\#\#\mathrm{Re}l < \#\#\overline{\mathrm{Re}l} \end{cases}, \tag{13}$$

where $\#Rel$ represents the number of $d_{j,i}$ whose relevance judgments equals to 1 in S_i for all S_i where d_j occurs and $\#\overline{Rel}$ represents the number of $d_{j,i}$ whose relevance judgments are 0 in S_i for all S_i where d_j occurs. An illustrative example is shown in Table 6 for the query "*true grit*". For example, for the candidate date "*2010*", $\#Rel = 7$ and $\#\overline{Rel} = 1$. As such $\left(q, d_j\right) = 1$.

In summary, our collection consists of 235 $\left(q, d_j\right)$ pairs, 86 labelled as incorrect or non-relevant (class 0) and 149 labelled relevant (class 1).

[13] https://www.microsoft.com/cognitive-services/en-us/bing-web-search-api [December 22, 2016].

Table 6 (q, d_j) classification for the query "*true grit*"

q	d_j	Id of S_i	$d_{j,i}$ judgment	(q, d_j) class
True Grit	1968	0, 6, 15, 47, 48	1, 1, 1, 1, 1	1
	1969	4, 6, 9, 27	1, 1, 1, 1	1
	1982	22	0	0
	2006	14	0	0
	2010	0, 1, 3, 12, 15, 24, 25, 29	1, 1, 1, 0, 1, 1, 1, 1	1
	2011	5, 37	0, 0	0

5.1.2 WC_TREC_DS collection

We further developed a new dataset to support our experiments over temporal information extracted from web contents. *WC_TREC_DS* dataset was created by selecting the 25 time sensitive queries from TREC-ts-{2013, 2014} (Guo et al. 2013) collections. We then queried the Bing search engine for each of the queries through Bing Search API and used Diffbot Article API[14] to collect the full text of the web page. We ended up with 489 texts and 443 distinct (q, d_j) pairs, where q is the query and d_j the candidate year. A list of all the queries is provided in Table 7.

The ground truth was then obtained by manually labeling each one of the 443 distinct (q, d_j) pairs. For this purpose, three human annotators were asked to consider each query, to look at the web search results and to assign each candidate date a relevance label on a 2-level scale: (0) not a date or temporally non-relevant to the query; (1) temporal relevant to the query.

The assessments were performed on November 2016 using Google Forms[15] and did not involve any payment. Each annotator evaluated 443 (q, d_j) pairs resulting in 1329 (q, d_j) total assessments, taking 3 h on average to complete their task. To get familiar with the topic, annotators were given a very short description of the query. The decision of whether a candidate date is relevant or not should take into account not only this short information, but also the web texts containing the candidate date. Thus, annotators are asked to not only determine the relevance of the obvious date, but also those candidate dates which despite being less evident may still be related to the query.

The final classification of each query comes by majority voting. As such, each candidate date is considered to be relevant to the query if it gets more relevant votes from the annotators than non-relevant ones. An inter-rater reliability analysis using the Fleiss Kappa statistics (1971) was then performed to determine consistency among annotators. Results have shown a value of 0.825, thus indicating an almost perfect agreement between the raters. The resulting ground-truth consists of 443 candidate dates, of which 194 were deemed relevant to the query (score 1) and 249 non-relevant (score 0).

[14] http://www.diffbot.com [December 22, 2016].

[15] http://bit.ly/2gk5DXX [December 22, 2016].

Table 7 List of text queries

buenos aires train crash	tel aviv bus bombing	colorado shooting	sikh temple shooting	hurricane isaac
pakistan factory fire	midwest derecho	typhoon bopha	guatemala earthquake	hurricane sandy
in amenas hostage crisis	european cold wave	queensland floods	costa concordia	egyptian riots
quran burning protests	boston marathon bombing	russian protests	romanian protests	egyptian protests
Southern California shooting	bulgarian protests	shahbag protests	nor'easter	russia meteor

5.2 Results and discussion

In this section, we discuss the results obtained from the experiments carried out. Our purpose is twofold. In the first set of experiments, we are particularly interested in studying how the temporal similarity measure GTE behaves against the baseline similarity measures. Our second objective is to evaluate the performance of the date filtering GTE-Class proposal against a number of machine learning methods. Each experiment will be conducted on top of the two collections previously introduced. A description of both experiments is given in Sects. 5.2.1 and 5.2.2 respectively. A summary of the results is presented in Sect. 5.2.3.

5.2.1 Temporal similarity measure (GTE) results

In order to assess the effectiveness of the GTE approach outlined in Sect. 4.3, we consider a number of baselines, both with temporal and non-temporal nature. For the **non-temporal** ones, we focus on considering pure corpus-based similarity measures which are characterized by not requiring access to external knowledge databases. We divide them into two groups: those based on word co-occurrences, and those based on web hit counts. The Pointwise Mutual Information (*PMI*) (Church and Hanks 1990), the Dice coefficient (Dice 1945), the Jaccard coefficient (1901) and the Symmetric Conditional Probability (*SCP*) (Silva et al. 1999) constitute the first group. While PMI tends to favor rare co-occurrences, SCP, DICE and Jaccard give more importance to more frequent co-occurrences. These measures are defined in Eqs. (14), (15), (16) and (17) respectively, where $P(x,y)$ corresponds to the joint probability that terms x and y co-occur in the same document, and $P(x)$ and $P(y)$ respectively correspond to the marginal probabilities that terms x and y appear in any document for a given query q:

$$PMI(x,y) = \log_2\left(\frac{P(x,y)}{P(x).P(y)}\right). \tag{14}$$

$$DICE(x,y) = \frac{2 \times P(x,y)}{P(x) + P(y)}. \tag{15}$$

$$Jaccard(x,y) = \frac{P(x,y)}{P(x) + P(y) - P(x,y)}. \tag{16}$$

$$SCP(x,y) = \frac{P(x,y)^2}{P(x) + P(y)}. \tag{17}$$

The other five similarity measures rely on the web as a corpus, by computing co-occurrences based on hit counts. This includes the Normalized Google Distance (*NGD*) (Cilibrasi and Vitányi 2007) and four other measures collected by Bollegala et al. (2007): WebJaccard, WebOverlap, WebDice and WebPMI. These are defined in Eqs. (18), (19), (20), (21) and (22) respectively. N is an estimation of the number of pages indexed by a given search engine which in the case of Google is near to 10^{10}, $h(x,y)$ returns the number of hits for the query "x y", $h(x)$ returns the number of hits for the query "x" and $h(y)$ returns the number of hits for the query "y":

$$NGD(x,y) = \frac{\max[\log h(x), \log h(y) - \log h(x,y)]}{\log N - \min[\log h(x), \log h(y)]}. \tag{18}$$

$$WebJaccard(x,y) = \frac{h(x,y)}{h(x) + h(y) - h(x,y)}. \tag{19}$$

$$WebOverlap(x,y) = \frac{h(x,y)}{min(h(x), h(y))}. \tag{20}$$

$$WebDICE(x,y) = \frac{2h(x,y)}{h(x) + h(y)}. \tag{21}$$

$$WebPMI(x,y) = \log_2\left(\frac{N.h(x,y)}{h(x) \times h(y)}\right). \tag{22}$$

To compare our approach over related work we further consider two additional baselines (Strötgen et al. (2012) and Kanhabua and Nørvåg (2010)) that make use of **temporal** signals. For the work of Strötgen et al. (2012) we rely on a set of heuristics that make it possible to determine the relevance of a temporal expression for a document with respect to a search query. Strötgen et al. (2012) do this by considering two relevance factors. The first one calculates the relevance of a temporal expression d_j with regards to information extracted from a particular document.

$$rel_g(d_j) = ((tf) + (sentLen) + (posSent) + (occType) + (sent) + (gran) + (tfIdf)) \tag{23}$$

where *tf* is the term frequency of the date in the document, *sentLen* is the length of the sentence in which the date occurs, *posSent* is the position of the date in the sentence, *occType* is the type of date occurrence in the document (in our case explicit temporal expressions), *sent* is the number of temporal expressions occurring in the same sentence, *gran* is the granularity of the date (in our case years), and *tfIdf* is the term frequency—inverse document frequency.

For the second factor the relevance of the temporal expression is calculated by taking into account its relationship with the query. With regard to this, four other functions are considered, *tqMatch* which infers if the date is within the interval expressed in the query (thus it only applies to explicit temporal queries), *tqDist* which measures the distance between the date found in the document and the date expressed in the query (similarly to the previous one it is only applicable to explicit temporal queries), *ttqSent* which describes if the date occurs in the same sentence as the query text, and *ttqDist* which returns the minimum distance in tokens between the date the query text part. Each of these functions are then given a value by means of pre-defined heuristics.

The computation of a single score comes as a combination between the two factors and is calculated as follows:

$$rel\left(q|d_j\right) = rel_g\left(d_j\right) \times ((tqMatch) + (tqDist) + (ttqSent) + (ttqDist)) \tag{24}$$

Note that this method is particular tuned to compute the relevance of a date with regard to a particular document. An average of the scores of the relevance of the date for the set of documents where it appears should thus be computed in order to determine its relevance with regards to the query.

We also compare our work against Kanhabua and Nørvåg (2010) a metadata-based dependent approach which builds upon temporal language models to compute statistics of word usage in all time intervals. Our temporal corpus is based on the set of documents retrieved for all the queries of our dataset, thus guaranteeing they cover the time period of the query. In spite of uniquely considering the publication date as a temporal clue, we consider all the dates found within the text contents, thus gathering a larger set of dates. The similarity score between the query (q) word and each of the time partitions (d_j) of the temporal language model (C) is then computed using a normalized log-likelihood ratio according to Eq. 25:

$$Score\left(q, d_j\right) = \sum_{w \in q} P(w|q) \times log\frac{P(w|d_j)}{P(w|C)}. \tag{25}$$

In our first experiment, which runs on top of the WC_DS collection, we compared several versions of the GTE combined with the InfoSimba (IS) similarity measure and the PMI, SCP and DICE similarity measures. Our aim is to understand its different behavior as PMI has often been preferred in the web context, as highlighted by Turney (Turney 2011).

Using InfoSimba (please recall Fig. 4) requires defining a size N for for both X and Y context vectors and a threshold similarity value, TH, such that, only those values from M_{ct} with similarity value $> TH$ should be considered as possible terms for the context vector representation. For this, we limited the parameters within the ranges of $5 \leq N \leq +\infty$ and $0 < TH \leq 0.9$ and combined them as: {*TH0.0N5, TH0.0N10, TH0.0N20, TH0.0* $N + \infty$, *TH0.05N5, TH0.05N10, TH0.05N20, TH0.05* $N + \infty$,..., *TH0.9N5, TH0.9N10, TH0.9N20, TH0.9* $N + \infty$}. For example, *TH0.0N5* means that we are selecting as context vectors of $w_{\ell,j}$ and d_j, the 5 most weighted terms registered in M_{ct} with similarity value higher than 0, i.e., that co-occur at least one time with $w_{\ell,j}$ and d_j respectively. Instead, *TH0.0* $N + \infty$ would use all the terms with a similarity value higher than 0.

To find an optimal combination of both we evaluated the combination of each of the three different aggregation functions (Max/Min, Mean and Median, denoted *MM*, *AM* and *M*, respectively), each of the three similarity measures combined with the IS (PMI, DICE and SCP) and each of the five context vector representations ((*W*;*W*), (*D*;*D*), (*W*;*D*), (*D*;*W*), (*WD*;*WD*)). The different versions of the GTE combined with IS are represented as *IS_*(*X*;*Y*)*_S_F*, where (*X*;*Y*) means the representation type of the context vectors, *S* the similarity measure used in IS and F the aggregation function that combines the different similarity values (registered in M_{ct}) between $w_{\ell,j} \in W_j^*$ and d_j. Further experiments have been performed based on the IS measure combined with PMI, SCP and DICE, but this time without the use of any aggregation function, i.e., by exclusively taking into account query q and candidate date d_j and not their correlated words $w_{\ell,j} \in W_j^*$. Overall, all of these measures are denoted *IS_*(X;Y)*_S*.

To identify the best combination of parameters, we measure, for each query pair, the correlation agreement between the values produced by each of the measures and the human annotations. With that in mind, we use the point biserial correlation coefficient (Katzell and Cureton 1947) which particularly suits this task. This statistical correlation measure relates a numerical variable with a variable consisting of binary or dichotomous classifications. In our case, "1" represents a relevant date and "0" represents either a false or non-relevant date. High biserial correlation values indicate high agreement with human annotations.

Our results have shown that the best combination was achieved for *T0.05* $N + \infty$, with a correlation value of 0.80 for the Median function combined with the *IS_(WD;WD)_-DICE_M* similarity value as shown in Table 8. This combination is denoted *BGTE* (*Best Generic Temporal Evaluation*) for the remainder of this article.

A further analysis led us to conclude that the Median and the Mean approach, overall, offer the best results when compared to the Max/Min. Even though the Mean approach is sensitive to extreme values, its performance is quite similar to the Median function which suggests that the IS measure has a symmetric distribution. In contrast, the Max/Min approach performs worst. This was expected given the existence of an arbitrary threshold which causes dates to be incorrectly classified as non-relevant. It is worth noting that, irrespective of the approach, the best correlation values always occur with the IS measure as shown in Table 9. This supports the hypothesis that a second-order metric behaves better than a first order similarity one.

In the following discussion, we show the effect of increasing the threshold *TH*. Results presented in Table 10 for N $+ \infty$, show that, *TH*0.0, *T*0.05 and *TH*0.1 perform quite well. However, they tend to become worse as *TH* gets increased. This is not surprising since increasing *TH* implies a sharp reduction of the number of possible candidates for each of the two context vectors, $w_{\ell,j}$ and d_j, as only relevant words and candidates dates that often co-occur with $w_{\ell,j}$ and d_j, will be considered.

While this guarantees that the two context vectors have strongly related tokens, it will naturally cause IS to perform worse. This is due to the lack of vocabulary, thereby decreasing the possibility of finding two tokens that co-occur at least once within the set of all web documents. Indeed, we may have a pair of words w_1 and w_2 which are strongly

Table 8 Point biserial correlation coefficient

Measure	Point Biserial	Measure	Point Biserial	Measure	Point Biserial
BGTE	**0.800**	Web Jaccard	−0.110	PMI	−0.0301
NK	0.301	Web Overlap	−0.060	SCP	0.358
JS	−0.063	Web Dice	−0.002	DICE	0.384
NGD	−0.065	Web PMI	−0.081	Jaccard	0.366

Bold value indicates the highest point biserial correlation value

BGTE versus Baselines. *T0.05* $N + \infty$

Table 9 Best point biserial correlation coefficient for GTE

Aggregation Function	Measure	*T0.05 N* $+ \infty$
Max/Min	IS_(WD;WD)_SCP_MaxMin	0.713
Mean	IS_(WD;WD)_DICE_Mean	0.799
Median	IS_(WD;WD)_DICE_Median	**0.800**

Bold value indicates the highest point biserial correlation value

correlated with $w_{\ell,j}$ and d_j respectively and yet IS will return a value of 0, as they never co-occur between them. A representation of this is given in Fig. 7.

It is also worth to note that the best biserial values often occur for *N20* and $N + \infty$ as opposed to $N5$ and $N10$. Once again, this shows that IS performs better when its context vectors contain a considerable number of tokens, as long as they guarantee a minimum value of co-occurrence with $w_{\ell,j}$ and d_j respectively. All the results are summarized in Fig. 8, for the three different approaches, when $5 \leq N \leq +\infty$ and $0 < TH \leq 0.9$.

A further observation led us to conclude that the type of the context vector representation greatly influences the performance of the system. We found that, regardless of the approach, the best possible representation is given by the combination of words and candidate dates, denoted (*WD*;*WD*). This was somehow expected inasmuch (*WD*;*WD*) gathers all the information available. Note however that a representation made only of words is also likely to reach good results as (*W*;*W*) was able to achieve quite similar results when compared to (*WD*;*WD*). An overall analysis of the results is given in Table 11 for *TH0.05 N* $+ \infty$.

Using the knowledge achieved, we then decided to test our system under a different collection of data, in particular on top of the WC_TREC_DS collection. With this new experiment, we aim to show that: (1) our system is not limited to a single query set; (2) its effectiveness does not depends on the size of the collection or data distribution; and (3) that it works well on top of any document, be it a full text or a snippet generated one.

For this purpose, we compared the BGTE against the NK, JS, Jaccard, Dice, PMI and SCP and left out the web-based measures which have proven not to suit this kind of task. We then used again the biserial correlation coefficient (Katzell and Cureton 1947) to identify the correlation agreement between the values produced by each of the measures and the human annotations.

Our results (see Table 12) have shown that the BGTE measure has once again performed better than any other baseline, achieving a 0.793 point biserial correlation value which is in line with the results obtained for the WC_DS collection.

Interestingly, all the baselines performed better when compared to the results obtained in our previous experiment. This is particularly evident for the PMI measure, which showcase an impressive increasing by going from −0.03 to −0.531 point biserial correlation value, meaning that it suits better on top of bigger texts. A detailed analysis of the results reveals however, that the PMI measure performance is in part, artificially boosted by the fact that larger texts will naturally tend to gather a higher number of d_j candidate date instances (e.g., *1412*) which occur only once in the text, but none of the times with its corresponding query (e.g., *colorado shooting*). This results in the occurrence of a considerable number of zero PMI values, the large majority of which, coinciding by chance with the non-date or non-relevant dominant class. Indeed, if we opt to remove all those (q, d_j) pairs for which the PMI value is zero, we end up with only −0.238 point biserial

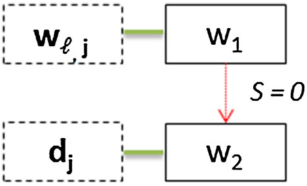
Table 10 Point biserial correlation coefficient for GTE

Aggregation Function	*TH0.0*	*TH0.05*	*TH0.1*	*TH0.2*	*TH0.3*	*TH0.4*	*TH0.5*	*TH0.6*	*TH0.7*	*TH0.8*	*TH0.9*
Max/Min	0.703	**0.713**	0.712	0.703	0.683	0.672	0.607	0.517	0.395	0.288	0.128
Mean	0.795	**0.799**	0.793	0.710	0.719	0.646	0.497	0.375	0.266	0.198	0.148
Median	0.799	**0.800**	0.788	0.668	0.710	0.632	0.474	0.329	0.156	0.094	0.085

Bold value indicates the highest point biserial correlation value for each one of the three aggregation functions

$0 < TH \leq 0.9$. N is fixed to $+\infty$

Fig. 7 $IS(w_j, d_j) = 0$



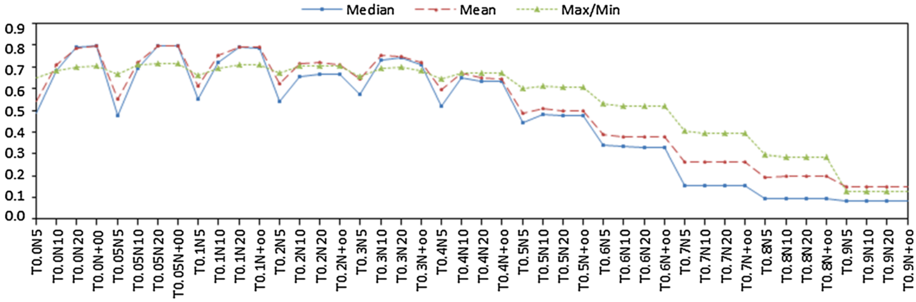


Fig. 8 Size and threshold effect. Median, Mean and Max/Min approach. Point biserial correlation values

Table 11 Best point biserial correlation coefficient for the five context vectors

Aggregation Function	(W;W)	(D;D)	(W;D)	(D;W)	(WD;WD)
Max/Min	0.706	0.545	0.333	0.449	**0.713**
Mean	0.768	0.358	0.387	0.149	**0.799**
Median	0.771	0.334	0.366	0.175	**0.800**

Bold value indicates the highest point biserial correlation value for each one of the three aggregation functions

TH0.05 $N + \infty$

Table 12 Point biserial correlation coefficient. BGTE versus Baselines

Measure	Point Biserial	Measure	Point Biserial
BGTE	**0.793**	PMI	−0.531
NK	0.422	SCP	0.280
JS	0.266	DICE	0.489
		Jaccard	0.462

Bold value indicates the highest point biserial correlation value for each one of the three aggregation functions

correlation, while still 0.714 for BGTE, 0.364 for the NK measure or even 0.355 for the jaccard, 0.371 for Dice and 0.298 for SCP, which, despite suffering from the same effects,[16] are not that exposed to this problem as PMI is. This further confirms that having a measure which is solely dependent on the co-occurrence of the query q and the candidate date d_j, which may never occur and yet be related, is too limited to determine their relevance. In this case, a second-order similarity measure is preferable. The provided

[16] Due to the similarity of the equations.

results also support the claim that the BGTE measure behaves well independently of the size of the collection, thus making it a good solution both for small or bigger texts.

We further perform an additional experiment to better understand the strengths and weaknesses of any of the measures being evaluated. We rest on Reciprocal Rank (see Eq. 26) to measure the rank at which the most obvious date is retrieved and use R-Precision (see Eq. 27) to measure the fraction of relevant dates for the query q that are successfully retrieved at the Rth position in the ranking list of results, where R is the total number of relevant dates for the query. The Mean R-Precision (MRP) and the Mean Reciprocal Rank (MRR) are then computed by taking the corresponding arithmetic mean of all the R-Precision and Reciprocal Rank values for the set of all the queries.

$$Reciprocal\,Rank(q) = \frac{1}{\mathrm{rank_q}}, \tag{26}$$

$$R\mathrm{-Precision(q)} = \frac{\mathrm{\#Rel_R}}{\mathrm{R}}, \tag{27}$$

Our results are shown in Fig. 9. The left-hand side concerns the MRR values, while the right one the MRP ones. Some key findings are that the NK measure achieves the best result on MRR with a small gain over BGTE, meaning it behaves better when detecting the most obvious date. However, no statistically significant differences between the retrieval effectiveness of the two methods were found, using a matched paired one-sided t test (p-values < 0.05). Another thing that stands out from our empirical evaluation is that the BGTE measure performs extensively better than any other approach when considering the retrieval of all the relevant dates for a query, as confirmed by the results obtained on the Mean R-Precision. Again, we measured statistical significance using a matched paired one-sided t-test with $p < 0.05$. The results obtained show that our measure outperforms all the baselines with a statistically significant difference. This is of particular interest as our aim is to retrieve not only the obvious points in time but also the set of correlated dates.

5.2.2 Date filtering (GTE-Class) results

The following experiment evaluates the performance of the date filtering proposal under two different collections, namely WC_DS and WC_TREC_DS. Our aim is to determine whether a date is relevant or not, as the simple identification of a year pattern is not enough to determine this. We use two different collections in order to understand whether there is any difference between using a small or a large corpus. To accomplish this objective we rely on the $BGTE(q, d_j)$ value to define a classical threshold-based strategy, where a

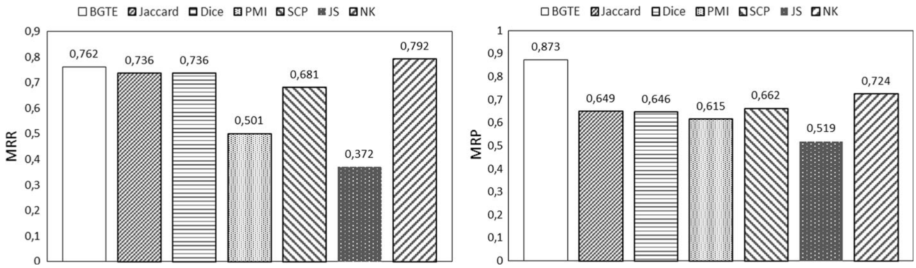


Fig. 9 MRR and MRP. BGTE versus Baselines

candidate date is deemed to be relevant if $BGTE(q, d_j)$ is above a given threshold (λ) or non-relevant otherwise. In order to determine the best λ, we rely on classical IR metrics, based on a confusion matrix with TP being the number of years correctly identified as relevant, TN being the number of years correctly identified as non-relevant or incorrect, FP being the number of years wrongly identified as relevant and FN being the number of years wrongly identified as non-relevant. Based on this, we calculate *Precision* (*P*), *Recall* (*R*), *Accuracy* (*A*) and *F1-Measure* (*F1-M*). To avoid overfitting and understand the generalization of the results, we followed a stratified *10*-fold cross-validation approach with 10 repetitions using the Weka implementation (Witten and Frank 2005) and the J48 decision tree algorithm. The values obtained for the BGTE measure point to 0.90 Accuracy, 0.93 Precision, 0.90 Recall and 0.92 F1-M for the WC_DS dataset, and 0.84 Accuracy, 0.85 Precision, 0.89 Recall and 0.87 F1-M for the WC_TREC_DS dataset when $\lambda = 0.35$. Figure 10 plots the obtained results for recall, precision and F1-M for both datasets. The dashed arrow is the λ threshold learned value that best optimizes the results when filtering out non-relevant dates.

To better understand the merits of our proposal we performed a further set of experiments. In our first experiment, we compare our proposal against each one of the top-3 similarity measures (Dice, Jaccard and NK) having achieved the highest biserial correlation coefficient, MRR and MRP. For each one of these measures, we then employ a classical threshold-based strategy to learn the relevance of the candidate dates, where each (q, d_j) pair is represented by its corresponding $sim(q, d_j)$ value, where $sim \in \{Dice, Jaccard, NK\}$. A stratified 10-fold cross-validation approach with 10 repetitions using the Weka implementation (Witten and Frank 2005) and the J48 decision tree algorithm was then applied, likewise in our approach. The final overall results of this experiment can be found on Table 13. All the results presented are statistically significant when comparing BGTE to the corresponding baseline methods with p-value < 0.05 using the matched paired one-sided t-test. As it can be noted BGTE clearly outperforms its corresponding baselines for all the IR metrics used which suggests that applying our model to a T-IR system will likely impact the effectiveness of the retrieved results. The experiment was also complemented with a Receiver Operating Characteristic (*ROC*) curve. The obtained results indicate BGTE as a good classifier with an AUC of 0.89 for the WC_DS dataset and 0.84 for the WC_TREC_DS dataset, high above the values obtained by all the other measures. The results of our experimental evaluation further confirm that λ is the same regardless the size of the collection. Another observation that stands out here is that although the effectiveness of the system decreases when comparing the results obtained for the WC_DS dataset and for the WC_TREC_DS one, this difference is marginal. The provided evidence supports the claim that the BGTE measure behaves mostly the same,

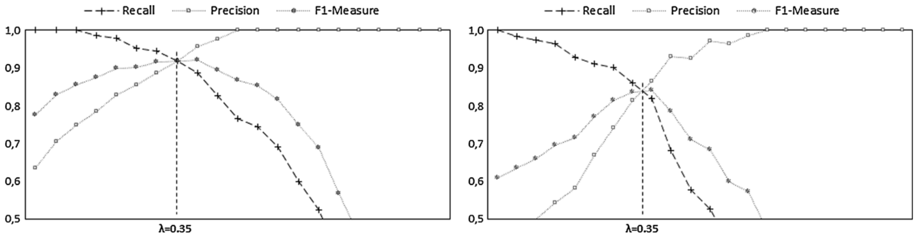


Fig. 10 Recall, Precision and F1-M performance when varying λ for the BGTE. The left-hand side refers to the WC_DS dataset and the right hand side to the WC_TREC_DS

Table 13 Comparative results for *sim*(*q*, d_j)

Measure	λ	Accuracy	Recall	Precision	F1-M	AUC
WC_DS						
BGTE	0.35	0.90	0.90	0.93	0.92	0.89
DICE	0.06	0.75	0.75	0.85	0.79	0.79
Jaccard	0.03	0.75	0.78	0.83	0.80	0.78
NK	0.18	0.74	0.71	0.87	0.77	0.76
WC_TREC_DS						
BGTE	0.35	0.84	0.89	0.85	0.87	0.84
DICE	0.10	0.70	0.85	0.70	0.77	0.69
Jaccard	0.05	0.70	0.85	0.70	0.77	0.69
NK	−5.12	0.67	0.97	0.64	0.77	0.63

either on small or large corpora, when filtering out non-relevant dates. Not surprisingly, the results of the NK measure are a little bit worst when compared to the BGTE approach. While the NK measure has obtained the best result for the detection of the most obvious date, the result obtained for the point biserial correlation coefficient was already indicating that this measure was not suitable for the kind of task we are dealing with in this experiment.

Next, we compare BGTE against two machine learning approaches, i.e., Kanhabua et al. (2012) and of Kawai et al. (2010), here denoted NK1 and HK respectively, which are specifically devoted to determine whether a candidate date is or not relevant. Likewise Strötgen et al. (2012), both methods are tuned to compute the relevance of a date with regard to a particular document (or sentence). In order to turn this into a query and date classification problem, an average of the values of features for the set of documents (or sentences) where the candidate date appears, is expected to be computed for each d_j. In the following, we detail the specificities of both works. For notational purposes, we refer to the work of Kanhabua et al. (2012) as NK1 and denote the work of Kawai et al. (2010) as HK.

Kanhabua et al. (2012) propose an approach to automatically identify relevant time for a set of events associated to a given named entity. In our approach, instead of named entities we assume queries as the main entry point and adapted their machine learning method on top of our collection. For this, a set of eleven sentence-based features were used to represent the triplet $\langle q, d_j, class \rangle$ in terms of a feature vector. Given a candidate date d_j, the values of the features are determined from the sentence Sen_k containing d_j. Next, we describe each one the features proposed and used in our experiments. The first feature *SenLen* is a score of the length (in characters) of Sen_k normalized by the maximum sentence length in the web text document T_i. The feature *isContext* indicates whether Sen_k contains the query. *cntQueryInS* is a score of the number of occurrences of the query in any sentence $Sen_k \in S_i$. The feature *cntTExpInS* is a score of the number of candidate dates d_j normalized by the maximum number of d_j in any sentence $Sen_k \in S_i$. The feature *QueryPos* is an average of scores of the positions (in characters) of the query in Sen_k normalized by the length of Sen_k. The feature *QueryPosDist* is an average of the scores of the position distance between all pairs of queries occurrence in Sen_k normalized by the length of Sen_k. The feature *TExpPos* is an average of scores of the positions (in characters) of candidates dates d_j in Sen_k normalized by the length of Sen_k. The feature *TExpPosDist* is an average of the scores of the position distance between all pairs of candidate dates in Sen_k normalized by the length of Sen_k. *timeDist* in turn is an average of scores of the distance in time for all pairs of candidate dates in Sen_k. Finally, the feature QueryTExpPosDist is an average of the scores of the position distance between all pairs of (q, d_j) in Sen_k normalized by the length of Sen_k.

Kawai et al. (2010) in turn propose a set of five features to represent the triplet $\langle q, d_j, class \rangle$. However, only three of them will be considered in our experiments as the remaining are either tailored to the Japanese language or dependent on their experimental collection. As the most basic feature, we implement unigrams (*UG*), i.e., nouns and verbs appearing with all the d_j candidates. A further feature is same window (*SW*) which refers to the distance between the query q and the candidate date d_j in $Sen_k \in S_i$. For this, a 3-term window was used to represent closeness. Finally, different year (*DF*) indicates if a different candidate date appears between the query q and the candidate date d_j in $Sen_k \in S_i$. The rationale is that if a further candidate date appears between both, the chance that a query q and the candidate date d_j are relevant would be lower.

Lay based on the authors experiments, we consider the use of all the features for each of the corresponding methods. In light of this, we then apply the J48 decision tree algorithm for NK1 and the SVM algorithm for HK, following the same parameters setting as before. As a further baseline, we also consider the majority classifier, which selects all of the candidate dates as relevant (i.e., recall = 1). All the results presented in Table 14 are statistically significant when comparing BGTE to the corresponding baseline methods with p-value < 0.05 using the matched paired one-sided t-test.

Again, we can conclude that the performance of the BGTE approach outperforms any of the baselines considered, regardless the size of the collection. Interestingly, we can also note that each one of the three similarity measures (DICE, Jaccard and NK) outperform any of the three machine learning methods considered (NK1, HK, Majority). Both results corroborate our research hypothesis which states that "The introduction of a classification model that is able to identify top relevant dates for any given implicit query while filtering out non-relevant ones, improves the correct classification of a query and a candidate date pair when compared to the baseline approach, which considers all the candidate dates as relevant for the query".

Table 14 Comparative results for BGTE versus machine learning approaches

Measure	Accuracy	Recall	Precision	F1-M	AUC
WC_DS					
BGTE	0.90	0.90	0.93	0.92	0.89
NK1	0.66	0.71	0.76	0.72	0.67
HK	0.59	0.78	0.66	0.69	0.54
Majority	0.63	1	0.63	0.77	0.50
WC_TREC_DS					
BGTE	0.84	0.89	0.85	0.87	0.84
NK1	0.72	0.60	0.72	0.64	0.74
HK	0.68	0.59	0.66	0.62	0.67
Majority	0.56	1	0.56	0.72	0.50

5.2.3 Summary

The experiments conducted above show that our proposal is capable of determining the most relevant dates related to a query when compared to different baseline measures. The results of our empirical evaluation show that:

- A combination of the second order similarity measure IS with the DICE coefficient and the Median aggregation function, denoted BGTE, leads to statistical significant improvement in evaluating the degree of relation between a query and a candidate date over all the other combinations;
- Our approach is able to identify top relevant dates for any given implicit temporal query in a large percentage of the cases, which is very promising given the complexity of the task. We do this based on a threshold classification strategy where $\lambda = 0.35$ was determined through a stratified *10*-fold cross-validation approach to enable a generalization of the results;
- Our results indicate that the difference between using the GTE-Class or any of the baseline methods is statistically significant for the correct classification of a $\left(\mathrm{q}, \mathrm{d_j}\right)$ pair.

The provided evidence clearly shows that the introduction of an additional layer of knowledge may affect the effectiveness of a broad set of T-IR systems, by retrieving a high number of precise relevant dates. This highlights the importance of considering temporal aspects in IR and the need for a continuous search for effective T-IR.

6 Conclusions and future work

Despite the fact that web documents contain many temporal expressions, few studies have fully used this information to improve web search diversity. Indeed, most of the IR systems do not yet incorporate temporal features in their architectures, treating all queries as if they were (temporally) equal. This limitation is due to the fact that retrieval models employed, continue to represent documents and queries rather simplistically, ignoring their underlying temporal semantics. Subsequently, they fail to understand the users' temporal intents.

The goal of this research was to design a model that tackles the temporal dimension of the user's queries, in order to identify the most relevant time periods. This demands not only the development of better document representations, which include temporal features, but also better temporal similarity metrics capable of reflecting the existing relation between the query and the set of extracted dates. In order to achieve this, we propose GTE, a new temporal similarity measure which allows employing different combinations of first order and second order similarity measures in order to compute the temporal intent(s) of $\left(q, d_j\right)$ pairs, plus GTE-Class a classification methodology (threshold-based), which is able to identify the set of top relevant dates for a given implicit temporal query, while filtering out the non-relevant ones.

In particular, we have shown that the combination of the second order similarity measure InfoSimba with the DICE coefficient and the Median aggregation function, denoted BGTE, leads to better results than all the other combinations and approaches, including temporal-based ones. Comparative experiments have been performed on two datasets which we made publicly available. Based on the results obtained we confirmed that our system behaves well independently of the size of the collection. We believe, that the introduction of an additional layer of knowledge may affect the effectiveness of a broad set of T-IR systems, by retrieving a high number of precise relevant dates.

Our data sets and experimental results are available online so that the research community can assess our results and propose new improvements to our methodologies. As an additional contribution to the research community, we publicly provide a number of web services and an online user search interface so that each of the different approaches can be tested. Although efficiency was not a core part of the framework, all the solutions perform quite well.

As future research, we aim to provide an effective clustering algorithm that clusters and ranks web documents, both based on their temporal and topical proximities. This can be further combined with a query temporal categorization strategy (Campos et al. 2011a) in order to boost more temporal clusters when the query is of temporal nature, while promoting more topical ones when the query is deemed to be atemporal. Names entities, when detected, could also be treated in a diverse way. Likewise, more relevant web documents to the query can be differentiated from those less relevant.

Note that the process of automatically deciding whether a date is or not relevant is always a risky procedure as it involves some kind of arbitrariness. Indeed, what is relevant to one user may not be to another. However, the simple fact that we could take into account the year's similarity relevance score with the query, is yet, per se, a major step when compared to the state of the art research. This could serve to improve the effectiveness of any IR system in several different tasks, such as query expansion, query reformulation, temporal clustering, temporal ranking, temporal result diversification or timeline design. Two demo applications involving temporal clustering (Campos et al. 2014b) and temporal ranking (Campos et al. 2014c) have already been presented in this scope.

Acknowledgements This research was funded by Project "TEC4Growth - Pervasive Intelligence, Enhancers and Proofs of Concept with Industrial Impact/NORTE-01-0145-FEDER-000020" which is financed by the North Portugal Regional Operational Programme (NORTE 2020), under the PORTUGAL 2020 Partnership Agreement, and through the European Regional Development Fund (ERDF) and by national founds of FCT-Foundation for Science and Technology under UID/MAT/00212/2013.

References

Alonso, O., Baeza-Yates, R., & Gertz, M. (2009). Effectiveness of Temporal Snippets. In *WWW'09 workshop on web search result summarization and presentation (WSSP'09)*. Madrid, Spain. April 20.

Alonso, O., Gertz, M., & Baeza-Yates, R. (2009). Clustering and exploring search results using timeline constructions. In *Proceedings of the 18th international ACM conference on information and knowledge management (CIKM'09)*. Hong Kong, China. November 2–6 (pp. 97–106).

Bollegala, D., Matsuo, Y., & Ishizuka, M. (2007) Measuring semantic similarity between words using web search engines. In *Proceedings of the 16th international world wide web conference (WWW'07)*. Banff, Canada. May 8–12 (pp. 757–766).

Brucato, M., & Danilo, M. (2014). Metric spaces for temporal information retrieval. In *Proceedings of the European conference on IR research (ECIR'14)*. Amsterdam, Netherlands. April 13–16 (pp. 385–397).

Campos, R., Dias, G., & Jorge, A. M. (2011a). What is the temporal value of web snippets? In *WWW'11 workshop on temporal web analytics (TWAW'11)*. Hyderabad, India. March 28.

Campos, R., Dias, G., Jorge, A. M., & Jatowt, A. (2014a). Survey of temporal information retrieval and related applications. *ACM Computing Surveys, 47*(2), 15.

Campos, R., Dias, G., Jorge, A. M., & Nunes, C. (2012). Enriching temporal query understanding through date identification: How to tag implicit temporal queries? In *WWW'12 workshop on temporal web analytics (TempWeb'12)*. Lyon, France. April 17 (pp. 41–48).

Campos, R., Dias, G., Jorge, A. M., & Nunes, C. (2014b). GTE-cluster: A temporal search interface for implicit temporal queries. In *Proceedings of the European conference on IR research (ECIR'14)*. Amsterdam, Netherlands. April 13–16 (pp. 775–779).

Campos, R., Dias, G., Jorge, A. M., & Nunes, C. (2014c). GTE-rank: Searching for implicit temporal query results. In *Proceedings of the 23rd ACM international conference on information and knowledge management (CIKM'14)*. Shanghai, China. November 3–7 (pp. 2081–2083).

Campos, R., Jorge, A., & Dias, G. (2011b). Using web snippets and query-logs to measure implicit temporal intents in queries. In *SIGIR'11 workshop on query representation and understanding (QRU'11)*. Beijing, China. July 28 (pp. 13–16).

Campos, R., Dias, G., Jorge, A., Nunes, C. (2016) GTE-Rank: A time-aware search engine to answer time-sensitive queries. *Information Processing & Management*, *52*(2), 273–298

Church, K. W., & Hanks, P. (1990). Word association norms mutual information and lexicography. *Computational Linguistics, 16*(1), 23–29.

Cilibrasi, R. L., & Vitányi, P. M. (2007). The google similarity distance. *IEEE Transactions on Knowledge and Data Engineering, 19*(3), 370–373.

Dakka, W., Gravano, L., & Ipeirotis, P. G. (2012). Answering general time sensitive queries. *IEEE Transactions on Knowledge and Data Engineering, 24*(2), 220–235.

Dias, G., Alves, E., & Lopes, J. (2007). Topic segmentation algorithms for text summarization and passage retrieval: An exhaustive evaluation. In *Proceedings of the 22nd conference on artificial intelligence (AAAI'07)*. Vancouver, Canada. July 22–26 (pp. 1334–1340).

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecological Society of America, 26,* 297–302.

Efron, M., & Golovchinsky, G. (2011). Estimation methods for ranking recent information. In *Proceedings of the 34th annual international ACM conference on research and development in information retrieval (SIGIR'11)*. Beijing, China. July 28 (pp. 495–504).

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological Bulletin, 76*(5), 378–382.

Foley, J., & Allan, J. (2015). Retrieving time from scanned books. In *Proceedings of the European conference on IR research (ECIR'15)*. Vienna, Austria. March 29–April 2 (pp. 221–232).

Guo, Q., Diaz, F., & Yom-Tov, E. (2013). Updating users about time critical events. Advances in information retrieval—Lecture Notes in Computer Science (Vol. 7814, pp. 483–494).

Gupta, D., & Berberich, K. (2014). Identifying time intervals of interest to queries. In *Proceedings of the 23rd ACM international conference on information and knowledge management (CIKM'14)*. Shanghai, China. November 3–7 (pp. 1835–1838).

Harris, Z. (1954). Distributional structure. *Word, 10*(23), 146–162.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura. *Bulletin de la Société Vaudoise des Sciences Naturelles, 37,* 547–579.

Jatowt, A., & Yeung, C. M. (2011). Extracting collective expectations about the future from large text collections. In *Proceedings of the 20th ACM conference on information and knowledge management (CIKM'11)*. Glasgow, Scotland, UK. October 24–28 (pp. 1259–1264).

Jatowt, A., Yeung, A.C.-M., & Tanaka, M. (2013). Estimating document focus time. In *Proceedings of the 22nd ACM conference on information and knowledge management (CIKM'11)*. San Francisco, USA. October 27–November 01 (pp. 2273–2278).

Joho, H., Jatowt, A., & Blanco, R. (2014). NTCIR temporalia: A test collection for temporal information access research. In *WWW'14 workshop on temporal web analytics (TempWeb'14)*. Seoul, Korea. April 8 (pp. 845–849).

Jones, R., & Diaz, F. (2007). Temporal profiles of queries. *ACM Transaction on Information Systems, 25*(3), 14.

Kanhabua, N., Blanco, R., & Matthews, M. (2011). Ranking related news predictions. In *Proceedings of the 34th annual international ACM conference on research and development in information retrieval (SIGIR'11)*. Beijing, China. July 24–28 (pp. 755–764).

Kanhabua, N., Blanco, R., & Nørvåg, K. (2015). Temporal information retrieval. *Foundations and Trends in Information Retrieval, 9*(2), 91–208.

Kanhabua, N., & Nørvåg, K. (2008). Improving temporal language models for determining time of non-timestamped documents. In *Proceedings of the European conference on research and advanced technology for digital libraries (ECDL'10)*. Aarhus, Denmark. September 14–19 (pp. 358–370).

Kanhabua, N., & Nørvåg, K. (2010). Determining time of queries for re-ranking search results. In *Proceedings of the European conference on research and advanced technology for digital libraries (ECDL'10)*. Glasgow, Scotland. September 6–10 (pp. 261–272).

Kanhabua, N., Romano, S., & Stewart, A. (2012). Identifying relevant temporal expressions for real-word events. In *SIGIR'12 workshop on temporal, social and spatially-aware information access (TAIA'12)*. Portland, USA. August 16.

Katzell, R. A., & Cureton, E. E. (1947). Biserial correlation and prediction. *The Journal of Psychology, 24*(2), 273–278.

Kawai, H., Jatowt, A., Tanaka, K., Kunieda, K., & Yamada, K. (2010). ChronoSeeker: Search engine for future and past events. In *Proceedings of the 4th international conference on ubiquitous information management and communication (ICUIMC'10)*. Suwon, Republic of Korea. January 14–15 (pp. 166–175).

Kulkarni, A., Teevan, J., Svore, K. M., & Dumais, S. T. (2011). Understanding temporal query dynamics. In *Proceedings of the fourth ACM international conference on web search and data mining (WSDM'11)*. Hong Kong, China. February 9–12 (pp. 167–176).

Li, X., & Croft, W. B. (2003). Time-based language models. In *Proceedings of the 12th ACM conference on information and knowledge management (CIKM'03)*. New Orleans, Louisiana, USA. November 2–8 (pp. 469–475).

Metzler, D., Jones, R., Peng, F., & Zhang, R. (2009). Improving search relevance for implicitly temporal queries. In *Proceedings of the 32th annual international ACM conference on research and development in information retrieval (SIGIR'09)*. Boston, USA. July 19–23 (pp. 700–701).

Moulahi, B., Lynda, T., & Sadok, B. Y. (2015). When time meets information retrieval: Past proposals, current plans and future trends. *Journal of Information Science*, *42*(6), 1–24.

Nunes, S., Ribeiro, C., David, G. (2008). Use of temporal expressions in web search. In *Proceedings of the European conference on IR research (ECIR'08)*. Glasgow, Scotland. March 30–April 3 (pp. 580–584).

Peetz, M.-H., Meij, E., & Rijke, M. (2014). Using temporal bursts for query modeling. *Information Retrieval Journal, 17*(1), 74–108.

Radinsky, K., Agichtein, E., Gabrilovich, E., & Markovitch, S. (2011). A word at a time: Computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on world wide web (WWW'11)*. Hyderabad, India. March 28–April 1 (pp. 337–346).

Ren, P., Chen, Z., Song, X., Li, B., Yang, H., & Ma, J. (2013). Understanding temporal intent of user query based on time-based query classification. In *Proceedings of the natural language processing and Chinese computing conference (NLPCC'13)*. Chongqing, China. November 15–19 (pp. 334–345).

Shokouhi, M., & Radinsky, K. (2012). Time-sensitive query auto-completion. In *Proceedings of the 35th annual international ACM conference on research and development in information retrieval (SIGIR'12)*. Portland, USA. August 12–16 (pp. 601–610).

Silva, J. F., Dias, G., Guilloré, S., & Pereira, J. G. (1999). Using local maxs algorithm for the extraction of contiguous and non-contiguous multiword lexical units. In *Proceedings of the 9th Portuguese conference in artificial intelligence (EPIA'99)*. Évora, Portugal. September 21–24 (pp. 21–24).

Strötgen, J., Alonso, O., & Gertz, M. (2012). Identification of top relevant temporal expressions in documents. In *WWW'12 workshop on temporal web analytics (TWAW'12)*. Lyon, France. April 17 (pp. 33–40).

Strötgen, J., & Gertz, M. (2015). A baseline temporal tagger for all languages. In *Proceedings of the 2015 conference on empirical methods in natural language processing (EMNLP'15)*. Lisbon, Portugal. September 17–21 (pp. 541–547).

Tran, G., Herder, E., & Markert, K. (2015). Joint graphical models for date selection in timeline summarization. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing of the asian federation of natural language processing (ACL'15)*. Beijing, China. July 26–31 (pp. 1598–1607).

Tran, G., Tran, T., Tran, N. K., Alrifai, M., & Kanhabua, N. (2013). Leveraging learning to rank in an optimization framework for timeline summarization. In *SIGIR 2013 workshop on temporal, social and spatially-aware information access (TAIA'13)*. Dublin, Ireland. August 1.

Turney, P. D. (2011). Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In *Proceedings of the 12th European conference on machine learning (EMCL'01)*. Freiburg, Germany. September 5–7 (pp. 491–502).

Vlachos, M., Meek, C., Vagena, Z., & Gunopulos, D. (2004). Identifying similarities, periodicities and bursts for online search queries. In *Proceedings of the international conference on management of data (ICMD'04)*. Paris, France. June 13–18 (pp. 131–142).

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques* (2nd ed.). San Francisco: Morgan Kaufmann.