Enriching Temporal Query Understanding through Date Identification: How to Tag Implicit Temporal Queries?

Ricardo Campos[1,2], Gaël Dias[4], Alípio Mário Jorge[1,3], Célia Nunes[5]
[1]LIAAD – INESC TEC
[2]Polytechnic Institute of Tomar, Portugal
[3]DCC – FCUP, University of Porto, Portugal
[4]HULTECH/GREYC, University of Caen Basse-Normandie, France
[5]Department of Mathematics and Center of Mathematics, University of Beira Interior, Covilhã, Portugal

ricardo.campos@ipt.pt, gael.dias@unicaen.fr, amjorge@fc.up.pt, celian@ubi.pt

ABSTRACT
Generically, search engines fail to understand the user's temporal intents when expressed as implicit temporal queries. This causes the retrieval of less relevant information and prevents users from being aware of the possible temporal dimension of the query results. In this paper, we aim to develop a language-independent model that tackles the temporal dimensions of a query and identifies its most relevant time periods. For this purpose, we propose a temporal similarity measure capable of associating a relevant date(s) to a given query and filtering out irrelevant ones. Our approach is based on the exploitation of temporal information from web content, particularly within the set of *k-top* retrieved web snippets returned in response to a query. We particularly focus on extracting years, which are a kind of temporal information that often appears in this type of collection. We evaluate our methodology using a set of real-world text temporal queries, which are clear concepts (i.e. queries which are non-ambiguous in concept and temporal in their purpose). Experiments show that when compared to baseline methods, determining the most relevant dates relating to any given implicit temporal query can be improved with a new temporal similarity measure.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *Query Formulation*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software – *Performance evaluation*

General Terms
Algorithms, Experimentation.

Keywords
Temporal Information Retrieval, Dating Implicit Temporal Queries, Temporal Query Understanding.

1. INTRODUCTION
Temporal Information Retrieval (T-IR) has been a topic of great interest in recent years. Its purpose is to improve the retrieval of documents by exploiting their temporal information, making it possible to position a topic in a timeline in accordance with its temporal dimension. Furthermore, it can also be very useful for other tasks, such as query understanding, query disambiguation, query classification, result diversification or even for relevance purposes. However and despite the relative maturity of the area and an increasing involvement of the IR community in recent years, few works have fully used temporal information for exploration and search purposes, as stated by Alonso et. al. [4].

TempWeb '12, Apr 16-17 2012, Lyon, France ACM 978-1-4503-1188-5/12/04.

Moreover, most of the existing approaches are built following the assumption that users always supply some form of temporal context, which is not always the case. Illustrative example of this are *FIFA World Cup Germany* and *George Bush Iraq War*. These are implicit temporal queries, which despite not containing an explicit temporal expression, may still have an inherent implicit temporal intent, {1974, 2006} in the former case and {1991, 2003} in the latter.

Understanding the timeline of this type of query is therefore of the utmost importance in order to improve the exploration of search results by means of temporal query expansion, timelines or temporal clustering. Nevertheless, there is little work addressing this particular problem and as referred by Berberich et. al. [6], it is therefore an interesting direction for future research. Moreover, none of the works extract temporal information from the contents of the document. They either follow a metadata-based approach that focuses on the publication date of the documents, or they follow a usage-based approach supported by web query logs. While the use of the former may be inadequate due to the fact that the time of a document may significantly differ from its actual content, query logs can prove to be very useful for user query understanding but very difficult to access outside of the big industrial labs. Furthermore, following a log-based approach to finding the temporal intents of implicit temporal queries, depends on the user's own intent and on the fact that some versions of the query have already been issued. Considering that only 1.21% of the queries include dates [9], this can be seen as a particular drawback.

Based on these statements and on the limitations of all of the above-mentioned approaches, we propose a web content approach. The extraction of temporal information within this context plays a fundamental role. Within the overall context of T-IR, most of the works rely on existing temporal annotation tools [1][19][30]. However, applying time-taggers to web collections based on simple regular expressions is likely to have a negative impact on the system's efficiency and effectiveness. This is mainly due to the simple fact that the identification of a year pattern is not enough to determine whether the date is real or whether it is query relevant. In this regard, note the following example (see Table 1) where we can see that neither *1500* is a date nor *2011* is relevant (as a date).

Table 1. Web snippet retrieved for the Haiti earthquake query.

Title	2011 Haiti Earthquake Anniversary
Snippet	As of 2010 (see 1500 photos here), the following major earthquakes have been recorded in Haiti. 1564 Haiti earthquake destroyed Concepción.

2. CONTRIBUTION AND OUTLINE

In this paper we are particularly interested in assessing the similarity between implicit temporal queries and dates (e.g., years). Specifically, we present a novel approach that aims (1) to correctly tag the temporal expressions found in the documents, based on their relevance to the query and (2) to properly tag implicit temporal queries with relevant years. Our method is not based on metadata or query-logs, but on the exploitation of temporal information from the text itself. In particular we use web snippets obtained from a web-search engine. As shown by Alonso et. al. [2] [5], this is an interesting alternative for the representation of web documents, providing a short summary of the document, where time clues, especially years, often appear [9]. Overall, we propose the following contributions in this paper:

1. We propose a novel second-order similarity measure to assess the temporal similarity between a query and a date based on a content-based language-independent approach;
2. We exhaustively evaluate our measure on a real-world dataset and demonstrate extensive improvements when compared to state-of-the-art techniques;
3. We publicly provide a set of queries and ground-truth results to the research community.

The remainder of this paper is organized as follows. Section 3 gives an overview of the related work. Section 4 formulates the problem to which we propose a possible solution. Section 5 introduces a new temporal similarity approach. Section 6 presents the details of our experiments. Section 7 provides the results of the experiments and finally, Section 8 concludes this paper and provides an outline of on-going work.

3. RELATED WORK

Within the overall context of T-IR, Jones et. al. [23] were the first to consider implicit temporal queries. They follow a metadata-based approach based on web news documents to model the period of time relevant to a given query by using a language model approach. Dakka et. al. [13] propose a similar solution, which takes into account the publication times of documents in order to favor more recent documents. Likewise, Kanhabua et. al. [24] propose three different methods to determine the time of queries using temporal language models, which are built based on the New York Times news collection, where documents are explicitly time-stamped according to the document creation time. Unfortunately, all of these works are language-dependent and mainly rely on the creation date of the documents as the correct temporal issue, which is far from true in most cases.

Finally, Metzler et. al. [28] propose mining query logs to identify implicit temporal information needs. They propose a weighted measure that considers the number of times a query, *q,* is pre- and post-qualified with a given year, *y*. A query is then implicitly year qualified if it is qualified by at least two unique years. Based on this, they propose a time-dependent ranking model that explicitly adjusts the score of a document in favor of those matching the user's implicit temporal needs. This work proposes an interesting solution as it introduces the notion of a correlation between a query and a year, but it lacks in query coverage as it depends on the analysis of query logs.

In this work, we propose a temporal similarity measure to associate relevant date(s) to a given query and filter out irrelevant ones based on a content-based approach. The method proposed makes use of the occurrences of words and dates in web snippets obtained from a web-search engine.

4. PROBLEM FORMULATION

This section first describes the problem of computing the similarity between a query and a date by specifying the key requirements for such a similarity measure. An algorithm is then presented to meet the proposed objective.

4.1 Problem Statement

Given a text query q, we obtain a collection of web snippets $S = \{S_1, S_2, \dots, S_n\}$ from n query related web pages. Each S_i, for $i = 1, \dots, n$, consists of its title and its snippet. It is worth noting that URLs are not considered since unlike titles and snippets, they embody noisy information, as shown in [9]. As such, S_i, for $i = 1, \dots, n$, is denoted as the concatenation of two texts, i.e. $\{Title_i, Snippet_i\}$, that can be represented by a bag-of-words and a bag-of-dates.

Specifically, W_{S_i} is defined as the set of the most relevant words/multi-words associated with a web snippet S_i. Moreover, the notation $W = \{w_1, w_2, \dots, w_p\}$ is used to define the set of the distinct most relevant words/multi-words extracted for the query, q, within the set of web snippets, S. Relevant words are collected based on the methodology proposed by Machado et. al. [26] who defines a heuristic based on the analysis of left and right word contexts. For convenience, throughout this paper, the most relevant words/multi-words will be referred to simply as words.

Similarly, D_{S_i} is defined as the set of years associated with a web snippet S_i. In addition, $D = \{d_1, d_2, \dots, d_m\}$ is defined as the set of distinct years extracted from the set of all web snippets S. In this approach, a simple rule-based model is used as proposed in [9] to extract explicit dates occurring within web snippets that satisfy certain specific explicit patterns, such as *yyyy*, *yyyy-yyyy*, *yyyy/yyyy*, *mm/dd/yyyy*, *mm.dd.yyyy*, *dd/mm/yyyy* and *dd.mm/yyyy*. It is important to note that our timeline is at the year granularity level, such that the extraction of the temporal expression *03/02/2009* will result in the year *2009*. Note that a document, as defined by Alonso et. al. [3], can also contain further types of temporal expressions other than explicit ones. They are known as implicit (e.g. *Christmas day*) and relative temporal expressions (e.g. *the next year*). Defining a set of regular expressions to capture them would, however, go in the opposite direction of our language independent goal. As such, they will not be studied in this paper.

Finally, W^* (formalized in equation 1) is defined as the set of distinct words that result from the intersection between the set of words W and the set W_{d_i} which contains the words that appear together with date, d_i, in any web snippet, S_i, from S.

$$W^* = W \cap W_{d_i} \quad (1)$$

Having defined the main concepts, it is now possible to formally define the problem.

The Problem Definition: given a query, q, issued by a user, and the set of dates, D, retrieved from the set of web snippets returned for the query q, assign a degree of relevance to each (q, d_i) pair. To model this relevance, a temporal similarity value, v, is defined given by a similarity measure sim.

$$v = sim(q, d_i)\ ,\ v \in [0,1]. \quad (2)$$

The proposed formulation attempts to identify relevant dates, d_i, for q and minimize any errors that might arise from considering irrelevant or wrong dates. The similarity value, v, is stored in a conceptual temporal correlation matrix defined as $M_{ct} = [CT_{ij}]_{p \times m}$, where CT_{ij} represents the similarity between the query, q, and the date, d_j. According to this research, it can be

concluded, however, that the relevance between a (q, d_i) pair is better defined if, instead of just focusing on the self-similarity, all of the information regarding existing temporal relations is augmented to a higher level, namely by calculating the similarities existing between W^* and d_i. Based on this, $sim(q, d_i)$ is defined on the assumption of the following principle.

Principle: The more closely a given date is correlated to the set of corresponding distinct most relevant words associated to the query (i.e. the intersection between the set of words co-occurring with the query and the set of words co-occurring with the date), the more closely the query will be associated to the date.

As such, part of the problem of this work will be to define not only the similarity between the query word and its respective temporal patterns but also between each of the most important topics present in web snippets and their respective temporal patterns. Based on this, the conceptual temporal correlation matrix is redefined as $M_{ct} = [CT_{ij}]_{p\times m}$, where CT_{ij} represents the similarity between the word, w_i, and the date, d_j. The following example represents the similarity values between a set of two words and three dates extracted from the set of web snippets, matching a dimension of 2 x 3, where for instance, 0.3 represents the conceptual temporal similarity between the word in the 1st row and the date in the 1st column.

$$M_{ct} = \begin{bmatrix} 0.3 & 0.2 & 0.8 \\ 0.1 & 0.4 & 0.6 \end{bmatrix}$$

While these similarity values can be determined by any similarity measure, either of first or second order, we believe they can be modeled more effectively if defined based on a second order attributional similarity measure. While for the first order association measures, the definition of the M_{ct} matrix is enough to compute $sim(q, d_i)$, as the relatedness between the query and the date is simply given by their co-occurrence, the use of a second-order co-occurrence measure is more complex. In fact, instead of simply relying on a direct co-occurrence, the definition of a second-order co-occurrence measure depends on setting a contextual vector for each of the two items (query and date). Each of the contextual vectors can be represented interchangeably by a set of words and a set of temporal patterns, such that the query and the date are similar if their contextual vectors are also similar. This requires the definition of two further matrices: a conceptual matrix denoted $M_c = [C_{ij}]_{p\times p}$, where C_{ij} represents the similarity between the word, w_i and the word, w_j and a temporal matrix denoted $M_t = [T_{ij}]_{m\times m}$ where, T_{ij}, represents the similarity between the date, d_i, and the date, d_j. Both matrices are symmetric with a diagonal equal to 1. This is clearly depicted in the following example, where 0.4 represents the conceptual similarity between the word in the 3rd row and the word in the 1st column (or the reverse) and 0.9 represents the temporal similarity between the date in the 3rd row and the date in the 1st column (or the inverse).

$$M_c = \begin{bmatrix} 1 & 0.8 & 0.4 \\ 0.8 & 1 & 0.1 \\ 0.4 & 0.1 & 1 \end{bmatrix} \quad M_t = \begin{bmatrix} 1 & 0.2 & 0.9 \\ 0.2 & 1 & 0.3 \\ 0.9 & 0.3 & 1 \end{bmatrix}$$

In summary, M_c, M_t and M_{ct} can be seen as the word-word, date-date and word-date matrices respectively containing the normalized similarity between all distinct relevant words, distinct years and distinct relevant words and distinct years, all identified from the set of web snippets for a given query. In addition, some key requirements that need to be fulfilled by the similarity measure are also formalized as follows:

R1: The more similar q and d_i are, the higher $v \in [0,1]$, with $sim(q, d_i)$ being close to 1 if d_i frequently co-occurs with W^*.

R2: d_i is more relevant for q than d_i', if $sim(q, d_i) > sim(q, d_i')$.

R3: d_i is more relevant for q than for q', if $sim(q, d_i) > sim(q', d_i)$.

R4: $sim(q, d_i) = 0$ if and only if d_i is not associated to any of the W^* words

4.2 The Algorithm

The framework of our query time tagging model is shown in Algorithm 1. The algorithm receives a query from the user, it fetches related web snippets from a given search engine and applies text processing to all web snippets. This processing task involves selecting the most relevant words and multi-words and collecting the years. Words and dates are then associated to a list of correlated distinct terms (e.g. words, multi-words or dates), where the correlation is based on web snippet co-occurrence. For the scoring phase all three matrices (conceptual, temporal and conceptual/temporal) are then computed. Finally, each date is given a temporal similarity value computed by a given sim similarity measure.

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

Input: query q
1: S ← RequestSearchEngine(q)
2: For each $S_i \in S$, i = 1,..,n
3: Apply Text Processing
4: W_{S_i} ← Select best relevant words/multi-words in S_i
5: D_{S_i} ← Select all temporal patterns in S_i
6: W ← Distinct words/multi-words in S
7: D ← Distinct temporal patterns in S
8: Compute M_c, M_t and M_{ct}
9: For each $d_i \in D$
10: Compute $sim(q, d_i)$
Output: (q, D) relevance

5. TEMPORAL SIMILARITY MEASURE

Most of the works in T-IR are based on the so called rule-based model. Under this model, all possible occurring dates are taken as relevant. As far as we know, the only work which presents a similarity measure for a query term and a year has been proposed by Metzler et al. [28] in the context of query logs. Unfortunately, it is not reproducible for web snippets. For that purpose, we propose a new generic temporal similarity measure called *GenTempEval*, which can be tuned for any association measures (of first or second order). The *GenTempEval* is formalized in equation 3, where sim is a similarity measure, and F is an aggregation function of the several $sim(W^*, d_i)$.

$$GenTempEval(q, d_i) = F(sim(W^*, d_i)), \qquad (3)$$

While sim can be any similarity measure, either of first or second order, it is evident that a second-order similarity measure carries additional valuable relations in both the query and the date contextual vectors, which cannot be induced if only a direct co-occurrence approach is used. In this context, most of the works apply the cosine similarity measure. However, as most of them rely on exact matches of context words, their accuracy is low since language is creative and ambiguous [18]. This is particularly evident in the case of relations between words and dates. In fact, depending on the context vector representation chosen, the cosine similarity measure may not even be applied. In order to overcome these drawbacks, other measures, such as Latent Semantic Analysis (LSA) [14] have been proposed. However, although LSA has shown interesting results in different areas [17], it has

also shown inefficiency when compared to other similarity measures, as highlighted by Turney et. al. [32].

This work applies the *InfoSimba* (*IS*) similarity measure proposed by Dias et. al. [15], a semantic vector space model supported by corpus-based word correlations. In detail, the *InfoSimba* calculates the correlation between two word vectors V_x and V_y, introducing into the Tanimoto Distance [29] a correlation factor between all word pairs existing between two word vectors. For this work, the *InfoSimba* is defined in Equation (4) where $S(i,j)$ is any first order similarity measure (e.g. DICE, PMI) relating words *i* and *j*, and $w(i)$ and $w(j)$ correspond to their weights, respectively.

$$IS(V_x, V_y) = \frac{\sum_{i \in V_x} \sum_{j \in V_y} w(i).w(j).S(i,j)}{\begin{pmatrix} \sum_{i \in V_x} \sum_{j \in V_x} S(i,j) + \\ \sum_{i \in V_y} \sum_{j \in V_y} S(i,j) - \\ \sum_{i \in V_x} \sum_{j \in V_y} S(i,j) \end{pmatrix}} \quad (4)$$

Without loss of generality, V_x and V_y can be seen as the context vector representation of each of the two items of a (q, d_i) pair, respectively. With this in mind, five possible representations were defined. The first one, denoted (*W*;*W*), represents both the query and the year through the vector of their correlated words registered in the M_c matrix. The second one, denoted (*D*;*D*), represents both the query and the year through the vector of their correlated dates, registered in the M_t matrix. The third one denoted (*W*;*D*) represents the query through the vector of its correlated words registered in the M_c matrix and the year through the vector of its correlated dates registered in the M_t matrix. The fourth representation here denoted (*D*;*W*) represents the query through the vector of its correlated dates registered in the M_{ct} matrix and the year through the vector of its correlated words registered in the M_{ct} matrix. Finally, the fifth representation denoted (*WD*;*WD*) represents both the query and the date through the vector of their correlated words and correlated dates, registered in the M_c, M_t and M_{ct} matrices, respectively.

Although *IS* has shown an improved performance compared to other state-of-the-art measures when directly applied to a (q, d_i) pair, results were not completely satisfactory. Motivated by our principle, we observed that the relevance between a (q, d_i) pair is defined better if, instead of just focusing on the self-similarity, all of the information regarding existing temporal relations is increased to a higher level, namely by calculating the similarities existing between W^* and d_i. Given this, we need an aggregation function *F* that combines the different similarity values produced for the date, d_i, in a single value capable of representing its relevance. For this purpose, we consider three different *F* functions, specifically (1) the Max/Min, (2) the Arithmetic Mean, and (3) the Median.

While the Mean and the Median methodologies are measures of central tendency, the Max/Min approach relies on the extreme values. In order to better understand this approach, we also define, in addition to R1 – R4, two other requirements, MAX and MIN.

MAX: the higher the number of relevant words related to the date, the higher the similarity. In detail, the system selects the maximum similarity value between the query and the year if the proportion of relevant words which appear with the date (i.e. W^*) is above a given threshold θ. In this case, θ has experimentally been defined as *0.2*.

MIN: the lower the number of relevant words related to the date, the lower the similarity. As such, proportion values ≤ 0.2 result in selecting the $sim(q, d_i)$ as a similarity value which is often the minimum one.

6. EXPERIMENT SETUP

Since no benchmark for (q, d_i) pairs exists, we built a new web-based dataset consisting of 582 web snippets and 235 distinct manually judged (q, d_i) pairs obtained from the execution of 42 real-world text clear-concept queries (i.e. non-ambiguous in concept). A number of experiments have been conducted on this collection. In particular, we assessed the performance of the new temporal similarity measure, as well as the effectiveness and efficiency in correctly classifying relevant dates.

6.1 Test Queries

In order to compare our approach with the baseline methods, we need a set of queries. Queries came from the 27 categories of Google Insights for Search Webpage [20]. We removed duplicates, discarded all concept ambiguous queries and applied the classification model proposed in Campos et. al. [10] to automatically select temporal queries. Our final collection consists of 42 representative clear-concept implicit temporal queries (i.e. non-ambiguous in concept and temporal in purpose).

6.2 Data Description

Based on the final list of 42 queries, we developed a web-content dataset (WC_DS) for our experiments which is publicly available for research purposes [8]. We queried the Bing search engine (*http://www.bing.com*) for each of the 42 queries, collecting the top best 50 relevant web results, with the en-US market language parameter, which resulted in a set of 582 relevant web snippets with years and 235 distinct (q, d_i) pairs. Each query had on average 14 corresponding related web snippets with an average of 1.2 years, of a total of 702 dates, 73 of which distinct. The ground truth was then obtained by manually labeling each one of the 235 distinct (q, d_i) pairs. Each one was assigned a relevance label by a human judge on a 2-level scale: not a date or irrelevant (score 0) and relevant date (score 1). In detail, we have 86 (q, d_i) pairs labeled with score 0, and 149 with score 1. The labeler was allowed to perform a search on the Web, so as to gain knowledge about the topic and eliminate context factors that might influence a change in his judgment.

6.3 Baseline Methods

In order to evaluate this approach, we will focus on corpus-based similarity measures as they are language-independent and do not require external knowledge. We compared several versions of the *GenTempEval* combined with the *InfoSimba* similarity measure, namely with three first order similarity measures (*PMI* [11], *SCP* [31] and *DICE* [16]). The aim was to understand its different behavior as the *PMI* has often been preferred in the web context as highlighted by Turney et. al. [32]. It is important here to note that the measure used within the *InfoSimba* is also used to select the most relevant words/dates for the context vectors in the five different possible context vector representations. Without losing generality, the different versions of the *GenTempEval* combined with *IS* are represented as *IS_(X;Y)_SM_F*, where *IS* stands for *InfoSimba*, (*X*;*Y*) means the type of contextual vector for both the query, q and the date, d_i, which are represented by their best co-occurring words and/or dates, *SM* is any similarity measure used in the *InfoSimba* and *F* is the aggregator function. Further experiments have been performed based on the *InfoSimba* measure combined with *PMI*, *SCP* and *DICE*, however this time without the use of any paradigm. Overall, all of these measures are denoted *IS_(X;Y)_SM*.

All other measures will be considered as state-of-the-art metrics. In particular, we will use the first order similarity measures (*PMI*, *DICE* , *Jaccard* [22], *SCP*) and web-based first order similarity

measures (*NgoogleDistance* [12], *WebJaccard* [7], *WebOverlap* [7], *WebDICE* [7], *WebPMI* [7]) with and without the aggregator function, denoted *SM* and *SM_F,* respectively.

The first four association measures are defined in equations (5), (6), (7) and (8), respectively, where $P(x,y)$ corresponds to the joint probability that words *x* and *y* co-occur in the same web snippet and $P(x)$ and $P(y)$ respectively correspond to the marginal probabilities that words *x* and *y* appear in any web snippet for a given query *q*. All other similarity measures are defined in equations (9), (10), (11), (12) and (13), respectively, where *N* is approximated by the number of pages indexed by a given search engine, which in the case of Google is near to 10^{10}, $f(x,y)$ returns the number of hits for query "*x y*" and $f(i)$ returns the number of hits for query "*i*".

$$PMI(x,y) = \log_2\left(\frac{P(x,y)}{P(x).P(y)}\right) \quad (5)$$

$$DICE(x,y) = \frac{2\times P(x,y)}{P(x)+P(y)} \quad (6)$$

$$Jaccard(x,y) = \frac{P(x,y)}{P(x)+P(y)-P(x,y)} \quad (7)$$

$$SCP(x,y) = \frac{P(x,y)^2}{P(x)+P(y)} \quad (8)$$

$$NGD(x,y) = \frac{\max[\log f(x),\log f(y)-\log f(x,y)]}{\log N-\min[\log f(x),\log f(y)]} \quad (9)$$

$$WebJaccard(x,y) = \frac{f(x,y)}{f(x)+f(y)-f(x,y)} \quad (10)$$

$$WebOverlap(x,y) = \frac{f(x,y)}{min(f(x),f(y))} \quad (11)$$

$$WebDICE(x,y) = \frac{2f(x,y)}{f(x)+f(y)} \quad (12)$$

$$WebPMI(x,y) = \log_2\left(\frac{N.f(x,y)}{f(x)\times f(y)}\right) \quad (13)$$

7. RESULTS AND DISCUSSION

Extensive experiments have been performed on a variety of measures using the WC_DS dataset in order to assess the performance of the three aggregator functions: Max/Min, Mean and Median, denoted *MM*, *AM* and *M*, respectively. It is worth recalling that the *GenTempEval* similarity measure can be tested over different association measures (of first and second order). Although its computation is direct for the first order metrics (equation 5 to equation 13), it requires certain configurations for the *InfoSimba* (equation 4). In this regard, we have already defined (1) the first order association measures to use with *IS* (*PMI*, *DICE* or *SCP*), and (2) the five possible context vector representations for the (q, d_i) pair: (*W*;*W*), (*D*;*D*), (*W*;*D*), (*D*;*W*), (*WD*;*WD*). Yet, it is still important to define the selection criterion from which to choose the set of words and/or dates that should be part of the (q, d_i) contextual vector representation. For this purpose, two inter-related factors should be considered: (1) the size of the contextual vector, denoted N, and (2) a threshold similarity value, T, from which, the terms that should be part of the contextual vectors, are selected. Based on this, we have performed a set of experiments with different sizes, N, and threshold values, T, for each of the three different approaches, in order to find an optimal combination of N and T. For this purpose, we have limited the parameters within the ranges of $5 \leq N \leq +\infty$ and $0 \leq T \leq 0.9$ and combined them as follows: {*T0.0N5, T0.0N10, T0.0N20, T0.0N+∞,..,T0.9N5, T0.9N10, T0.9N20, T0.9N+∞*}. The following sub-sections show the results of all of the experiments.

7.1 Temporal Similarity of a (q, d_i) pair

First, the results of the experiment showed that the best correlation coefficient, regardless of the approach used, was always *T0.05N+∞* (i.e. the selection of all terms *N+∞* with a similarity value $T > 0.05$). In particular, in order to assess the best tuned parameters for the WC_DS dataset, we used the point biserial correlation coefficient [25]. Unlike the Pearson measure, which is usually used in this case, the point biserial is a statistical correlation measure that considers items consisting of binary or dichotomous classifications (i.e., for which two possible answers are admitted). In this particular case, 1 represents a date and 0 represents either a false date or an irrelevant one. High correlation biserial values indicate high agreement with human annotators. A summary of the best results for the three different aggregator functions are shown in Table 2. The best correlation value (i.e. 0.80) is given for the Median function, specifically for *IS_(WD;WD)_DICE_M*, denoted *BGTE* (Best *GenTempEval*) in the remainder of this paper. Overall, the Median and the Mean give the best results when compared to the Max/Min. In particular, although they are sensitive to extreme values, the performance of the Mean approach is equal to the Median function, which suggests that the *IS* have a symmetric distribution. In contrast, the Max/Min approach performs worst. This was expected due to the existence of an arbitrary threshold, which causes dates to be incorrectly classified as irrelevant.

Table 2. Best Point biserial Correlation Coefficient for *GenTempEval*

Aggregation	Measure	N5	N10	N20	*N+∞*
Max/Min	*IS_(WD;WD)_SCP_MM*	0.668	0.708	0.712	**0.713**
Mean	*IS_(WD;WD)_DICE_AM*	0.550	0.724	0.795	**0.799**
Median	*IS_(WD;WD)_DICE_M*	0.540	0.693	0.795	**0.800**

It is worth noting that, regardless of the approach the best correlation values always occur with the *InfoSimba* measure. This supports the assumption that a second-order co-occurrence measure behaves better than a first-order similarity one. Moreover, the best results occur with a higher selection of terms (*N+∞*). This is particularly evident for the Median approach, where we can note a further improvement of 0.26 point biserial correlation, compared to $N5$.

Moreover, we can also note from Table 3, that the type of contextual vector representation chosen for the (q, d_i) pairs greatly influences the performance of the system. Regardless of the approach used, we found that the best possible representation is given by the combination of words and dates, denoted (*WD*;*WD*).

Table 3. Best Point biserial Correlation Coefficient for the 5 contextual vectors

Aggregation	(*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**

Table 4. List of (q, d_i) examples with the BGTE for the Median aggregator function compared to baseline methods.

(q, d_i) Pair	Class	BGTE	NGD	WebJaccard	WebDICE	WebPMI	PMI	DICE	Jaccard	SCP
True grit – 1969	1	0.896	0.360	0.290	0.012	0.325	0.378	0.255	0.194	0.217
True grit – 2010	1	0.812	0.327	0.336	0.201	0.414	0.378	0.750	0.679	0.759
Avatar movie – 2009	1	0.670	0.325	0.516	0.621	0.455	0.261	0.412	0.330	0.214
Avatar movie – 2011	0	0.346	0.330	0.454	0.515	0.432	0.261	0.102	0.074	0.043
California king bed – 2010	1	0.893	0.334	0.398	0.388	0.417	0.518	0.329	0.257	0.287
Slumdog millionaire – 2009	0	0.000	0.311	0.350	0.251	0.461	0.388	0.069	0.049	0.055
Tour Eiffel – 1512	0	0.286	0.331	0.288	0.001	0.267	0.432	0.075	0.054	0.060
Lady gaga – 1416	0	0.336	0.337	0.289	0.003	0.275	0.368	0.066	0.047	0.053
Haiti earthquake – 2010	1	0.605	0.328	0.339	0.210	0.426	0.449	1.000	1.000	1.000
Sherlock Holmes – 1887	1	0.839	0.342	0.292	0.020	0.330	0.388	0.135	0.099	0.111
Dacia duster – 1466	0	0.096	0.323	0.288	0.000	0.206	0.378	0.067	0.048	0.054
Waka waka – 1328	0	0.246	0.321	0.288	0.000	0.102	0.492	0.084	0.061	0.068
Waka waka – 2010	1	0.944	0.328	0.332	0.188	0.420	0.492	0.742	0.670	0.749
Bp oil spill – 2006	0	0.277	0.300	0.350	0.248	0.454	0.545	0.094	0.068	0.076
Bp oil spill – 2010	1	0.838	0.328	0.323	0.154	0.426	0.254	0.384	0.304	0.211
Volcano Iceland – 2010	1	0.749	0.000	0.288	0.000	0.290	0.368	0.000	0.000	0.000
Point Biserial Correlation	-	0.800	-0.065	-0.110	-0.002	-0.081	-0.031	0.385	0.366	0.358

Finally, in Table 4 we calculate the similarity scores between a set of (q, d_i) pairs to compare the *BGTE* with baseline methods and to show that all four requirements defined in section 4.1 are met. Similarity, scores are normalized into a range of [0..1] for ease of comparison. The bottom row of the table shows the point biserial correlation coefficient for each of the baseline methods. The highest correlation is reported by the proposed *BGTE* with a notable improvement shown when compared to all the other measures. It is also noteworthy that first-order measures, such as SCP, Dice and Jaccard outperform Web-based similarity ones. One reason for this is that Web-based measures offer limited reliability when estimating term correlation due to ambiguity and the non-existence of content analysis [7]. This is a problem that tends to get even worse in a temporal context.

Table 4 also shows that **R1** is taken into account by *GenTempEval*, as the similarity of the $(waka\ waka, 2010)$ pair is close to 1. Following our **Principle**, the final score is computed by considering all of the similarities between $(W^*, 2010)$ (e.g., [{*2010;fifa world cup song*},*0.922*], [{*2010;africa*}, *0.977*], [{*2010;shakira waka waka*}, *0.961*], and so on).

There are other examples in the table for which remaining formulations are clear, for example, d_i is more relevant for q than d_i' (**R2**). This can be demonstrated since the *BGTE* similarity score of *(avatar movie,2009)* is higher than the similarity rate of *(avatar movie, 2011)*. Similarly, **R3** is fulfilled since the date *2010* is more relevant to the query *waka waka* than to the query *bp oil spill*. Finally, if any relevant word is related to the date, $sim(q, d_i) = 0$. This requirement (**R4**) is fulfilled by the pair (*slumdog millionaire*, *2010*).

7.2 Relevance Classification of a (q, d_i) pair

To determine whether a date is relevant or not for a given query, we use a classical threshold-based strategy. Given a (q, d_i) pair, the system automatically classifies a given date as relevant or irrelevant based on two complementary conditions: (1) relevant, if $(GenTempEval(q, d_i) >= \lambda)$, and (2) irrelevant or wrong date, if $GenTempEval(q, d_i) < \lambda$.

In order to evaluate this strategy, we propose calculating the classical evaluation metrics in IR, which includes Precision, F1-Measure (denoted F1), Balanced Accuracy (or Efficiency), Recall (or Sensitivity), Specificity and Accuracy. A summary of the results of the experiments can be seen in Table 6 to Table 9. The values presented correspond to the best tuned λ for each one of the measures. In order to determine the best λ, we used the ROC curve. Figure 1 plots this curve for the *BGTE*. The red line indicates an almost perfect classifier with an Area Under Curve (AUC) of 0.960. The standard error of the curve is 0.011.

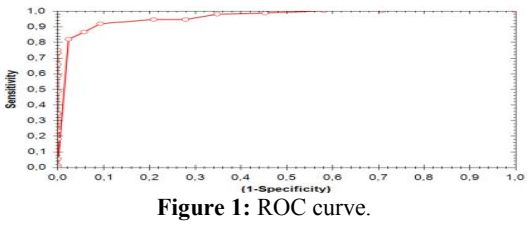

Figure 1: ROC curve.

The best optimization cutoff corresponds to the point that is closest to the upper left hand corner of the diagram, since the index of True Positives (*TP*) is one and the one of False Positives (FP) is zero. In the case of the *BGTE*, this corresponds to 0.093 of 1-Specificity and 0.919 of Sensitivity matching a cutoff of λ = 0.35 (see Table 7). In particular, increasing or decreasing λ affects Recall, Precision and *F1*-Measure, as illustrated in Figure 2.

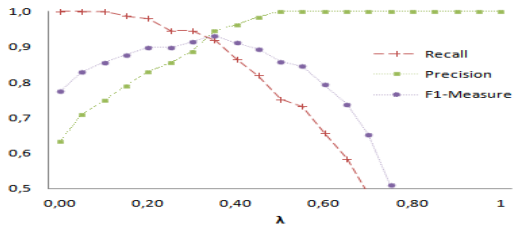

Figure 2: Performance Results vs λ.

From Table 7, we can also observe that BGTE can achieve a 93.1% F1 performance, 91.3% of Balanced Accuracy, 94.4% of Precision, 91.4% of Accuracy and a Recall of 91.9%. When compared to the best non-IS self-similarity method (see Table 6), which is *DICE*, the *BGTE* can produce 21.4% *F1* improvements. A general overview, for Recall, Precision and *F1* is also reproduced in Figure 3, which shows a clear difference between the *BGTE* and the results of the traditional state-of-the-art similarity measure.

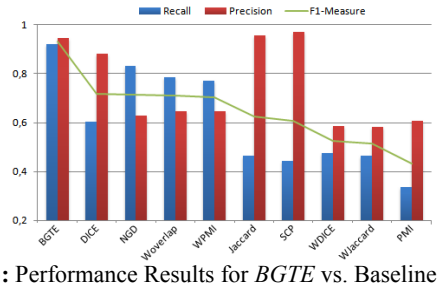

Figure 3: Performance Results for *BGTE* vs. Baseline Methods

A further observation shows that simply by adding the Median aggregator function (Table 7) to the simple *IS_(WD;WD)_DICE* (Table 6) results in an improvement of 9.8% in terms of F1. Indeed, all similarity measures within *GenTempEval* outperform their baselines in terms of *F1*, indicating that using the Median as part of the model positively impacts the performance of the system. Nevertheless, even by including the Median aggregator function with state-of-the-art measures, results are far from equaling the performance of the *BGTE*. In this case, the best measure is *Jaccard_M*, however it shows a negative difference of 8.4% *F1*, 15.4% Precision and an AUC of 21.1%.

Similar conclusions can be taken for the Mean approach (see Table 8). Indeed, when comparing the Median with the arithmetic Mean, the results favor the first methodology, although the differences are small. Finally, Table 8 shows the results for the Max/Min approach. Likewise the Median and the Mean aggregators, all of the measures under the Max/Min approach outperform their baselines (Table 6) in terms of *F1*. However, the *IS_(WD;WD)_DICE_MM* shows a decrease in performance of up to 5.4% *F1* when compared to the *BGTE*. Interestingly, all non-IS measures, except for the *WebOverlap*, yield the best *F1* results when compared to their corresponding aggregator measures, due to a strong improvement in terms of precision. Moreover, within this context, the Max/Min approach is the one that presents the best results.

7.3 Statistical Significance Tests

In our final experiment, we compare the results of the *BGTE's* performance with the baseline rule-based model (current standard in most of the T-IR tasks), which selects all of the temporal patterns found as correct dates (i.e. Recall = 1) within a given dataset. As a consequence, for a fair comparison we defined a Recall of 1 for the *BGTE*. Results are presented in Table 5. Although the *BGTE* threshold strategy is weakened by a recall equal to one, our methodology still significantly outperforms the baseline model.

Table 5. Performance Results: BGTE vs. Baseline

	Baseline	BGTE
Precision	0.634	**0.748**
Recall	1	1
F1-Measure	0.776	**0.856**

Furthermore, we assess if the difference between using the *BGTE* or Baseline for the correct classification of a (q, d_i) pair is significant. For this purpose, we performed a McNemar's test [27], a non-parametric method that is particularly suitable for testing dichotomous variables, which is this case. The test resulted in a Chi-squared statistic value equal to 126.130 with a p-value < 2.2e-16. Based on this result, we also built a confidence interval for the difference of means for paired samples between the number of misclassified dates given by the rule-based method and by the *BGTE*. The interval obtained [1.42; 2.30] clearly shows that the rule-based model retrieves, on average, more irrelevant or incorrect dates than the *BGTE* measure, with a 95% confidence level.

8. CONCLUSION AND FUTURE WORK

In this work we proposed a new temporal similarity measure, the *GenTempEval*, which makes it possible to test different combinations of first order and second order similarity measures in order to compute the temporal intent(s) of (q, d_i) pairs. While the techniques employed are not completely novel, they showed improved results when applied to this specific problem. In particular, we showed that the combination of the second order similarity measure *InfoSimba* with the DICE coefficient and the Median aggregator function shows better results than all other combinations based on a threshold classification strategy where λ = 0.35 has been automatically evaluated. We believe that the proposed method will be useful to disambiguate a large class of implicit temporal queries. As future work we plan to compare our approach with a query log based one and a temporal classifier based on multiple similarity measures. We also plan to use this new classifier in the field of Temporal Ephemeral Clustering. Indeed, as it is language-independent and does not depend on lists of stop-words, it can be applied in real-world search scenarios.

Table 6. Evaluation results on WC_DS dataset for the Simple approach: $sim(q, d_i)$

Measure	λ	TP	TN	FP	FN	1-Specificity	Recall	Precision	Accuracy	Bal.Accur.	F1	AUC	Error
IS_(WD;WD)_SCP	0.15	66	84	2	83	0.058	0.671	0.952	0.638	0.806	0.787	0.709	0.033
IS_(WD;WD)_DICE	0.15	113	77	9	36	0.104	0.758	0.926	**0.808**	**0.826**	**0.833**	**0.845**	**0.024**
IS_(WD;WD)_PMI	0.25	116	41	45	33	0.523	0.778	0.720	0.668	0.627	0.748	0.594	0.037
SCP	0.05	66	84	2	83	**0.023**	0.442	**0.970**	0.638	0.709	0.608	0.709	0.033
PMI	0.05	50	54	32	99	0.372	0.335	0.609	0.422	0.481	0.432	0.558	0.038
DICE	0.05	90	74	12	59	0.139	0.604	0.882	0.697	0.732	0.717	0.757	0.030
Jaccard	0.05	69	83	3	80	0.034	0.463	0.958	0.646	0.714	0.624	0.714	0.033
WebPMI	0.9	115	23	63	34	0.732	0.771	0.646	0.587	0.519	0.703	0.577	0.038
WebDICE	0.1	71	36	50	78	0.581	0.476	0.586	0.455	0.447	0.525	0.584	0.037
WebJaccard	0.05	69	37	49	80	0.569	0.463	0.584	0.451	0.446	0.516	0.589	0.037
WebOverlap	0.05	117	22	64	32	0.744	0.785	0.646	0.591	0.520	0.709	0.555	0.038
NGoogleDistance	0.75	124	13	73	25	0.848	**0.832**	0.629	0.582	0.491	0.716	0.508	0.039

Table 7. Evaluation results on WC_DS dataset for $F(sim(W^*, d_i))$, $F = Median$

Measure	λ	TP	TN	FP	FN	1-Specificity	Recall	Precision	Accuracy	Bal.Accur.	F1	AUC	Error
IS_(WD;WD)_SCP_M	0.25	137	63	23	12	0.267	0.919	0.856	0.851	0.826	0.886	0.909	0.018
IS_(WD;WD)_DICE_M	0.35	137	78	8	12	**0.093**	0.919	**0.944**	**0.914**	**0.913**	**0.931**	**0.960**	**0.011**
IS_(WD;WD)_PMI_M	0.15	148	31	55	1	0.639	0.993	0.729	0.761	0.676	0.840	0.682	0.034
SCP_M	0.05	134	25	61	15	0.709	0.899	0.687	0.676	0.595	0.779	0.549	0.038
PMI_M	0.1	147	19	67	2	0.799	0.986	0.686	0.706	0.603	0.809	0.597	0.037
DICE_M	0.15	144	34	52	5	0.604	0.966	0.734	0.757	0.680	0.834	0.648	0.036
Jaccard_M	0.1	136	50	36	13	0.418	0.912	0.790	0.791	0.747	0.847	0.749	0.031
WebPMI_M	0.6	145	9	77	4	0.895	0.973	0.653	0.655	0.538	0.781	0.519	0.039
WebDICE_M	0.65	74	47	39	75	0.453	0.496	0.654	0.514	0.521	0.564	0.516	0.039
WebJaccard_M	0	136	11	75	13	0.872	0.912	0.644	0.622	0.520	0.755	0.700	0.033
WebOverlap_M	0.95	96	28	58	52	0.674	0.644	0.623	0.527	0.484	0.633	0.519	0.039
NGoogleDistance_M	0.75	149	3	83	0	0.965	**1**	0.642	0.646	0.517	0.782	0.517	0.039

Table 8. Evaluation results on WC_DS dataset for $F(\mathbf{sim}(W^*, d_i))$, $F = Mean$

Measure	λ	TP	TN	FP	FN	1-Specificity	Recall	Precision	Accuracy	Bal.Accur.	F1	AUC	Error
IS_(WD;WD)_ SCP_AM	0,30	136	70	16	13	0,186	0,912	0,894	0,876	0,863	**0,903**	0,933	0,015
IS_(WD;WD)_DICE_AM	0,35	138	75	11	11	**0,127**	0,872	0,906	**0,899**	**0,926**	0,872	**0,963**	**0,011**
IS_(WD;WD)_PMI_AM	0,15	148	31	55	1	0,639	**0,993**	0,729	0,761	0,676	0,840	0,684	0,034
SCP_AM	0,05	146	21	65	3	0,755	0,979	0,691	0,710	0,612	0,811	0,606	0,037
PMI_AM	0,15	148	18	68	1	0,790	0,993	0,685	0,706	0,601	0,810	0,589	0,037
DICE_AM	0,15	148	29	57	1	0,662	0,993	0,721	0,753	0,665	0,836	0,695	0,034
Jaccard_AM	0,10	147	42	44	2	0,511	0,986	0,769	0,804	0,737	0,864	0,798	0,028
WebPMI_AM	0,45	121	12	74	28	0,860	0,812	0,620	0,565	0,475	0,703	0,593	0,037
WebDICE_AM	0,85	95	44	42	54	0,488	0,637	0,693	0,591	0,574	0,664	0,555	0,038
WebJaccard_AM	0,05	72	41	45	77	0,523	0,483	0,615	0,480	0,479	0,541	0,745	0,031
WebOverlap_AM	0,95	126	20	66	23	0,767	0,845	0,656	0,621	0,539	0,739	0,541	0,038
NGoogleDistance_AM	0,75	149	3	83	0	0,965	0,517	**1**	0,642	0,646	0,681	0,517	0,039

Table 9. Evaluation results on WC_DS dataset for $F(sim(W^*, d_i))$, $F = Max/Min$

Measure	λ	TP	TN	FP	FN	1-Specificity	Recall	Precision	Accuracy	Bal.Accur.	F1	AUC	Error
IS_(WD;WD)_ SCP_MM	0,55	122	75	11	27	0.127	0.818	0.917	0.838	0.845	0.865	0.883	0.021
IS_(WD;WD)_DICE_MM	0,7	122	79	7	27	**0.081**	0.818	**0.945**	**0.855**	**0.868**	**0.877**	**0.895**	**0.020**
IS_(WD;WD)_PMI_MM	0,2	128	66	20	21	0.232	0.859	0.864	0.825	0.813	0.861	0.858	0.023
SCP_MM	0,05	128	65	21	21	0.244	0.859	0.859	0.821	0.807	0.859	0.835	0.025
PMI_MM	0,2	128	65	21	21	0.244	0.859	0.859	0.821	0.807	0.859	0.799	0.028
DICE_MM	0,15	128	66	20	21	0.232	0.859	0.864	0.825	0.813	0.861	0.848	0.024
Jaccard_MM	0,1	128	66	20	21	0.232	0.859	0.864	0.825	0.813	0.861	0.842	0,024
WebPMI_MM	0,6	142	18	68	7	0.790	0.953	0.676	0.680	0.581	0.791	0.523	0.038
WebDICE_MM	0,75	103	63	23	46	0.267	0.691	0.817	0.706	0.711	0.749	0.732	0.032
WebJaccard_MM	0,6	95	65	21	54	0.244	0.637	0.818	0.680	0.696	0.716	0.724	0.032
WebOverlap_MM	0,95	127	38	48	22	0.558	0.647	0.633	0.725	0.702	0.640	0.649	0.035
NGoogleDistance_MM	0,05	143	12	74	6	0.860	**0.959**	0.658	0.659	0.549	0.781	0.549	0.038

9. ACKNOWLEDGMENTS

This work is funded by the ERDF through the Programme COMPETE and by the Portuguese Government through FCT - Foundation for Science and Technology and grant (Reference: SFRH/BD/63646/2009). It is also supported by the Center of Mathematics, University of Beira Interior, project PEst-OE/MAT/UI0212/2011.

10. REFERENCES

[1] ANNIE (2002). http://www.aktors.org/technologies/annie/

[2] Alonso, O., Baeza-Yates, R., and Gertz, M. (2009). Effectiveness of Temporal Snippets. In *WSSP'09 - WWW'09*. Madrid, Spain.

[3] Alonso, O., Baeza-Yates, R., & Gertz, M. (2007). Exploratory Search Using Timelines. In *ESCHI - CHI'07*. San Jose, USA.

[4] Alonso, O., Gertz, M., and Baeza-Yates, R. (2009). Clustering and Exploring Search Results using Timeline Constructions. In *CIKM'09*.

[5] Alonso, O., Gertz, M., and Baeza-Yates, R. (2011). Enhancing Document Snippets Using Temp. Information. LNCS 7024, 26-31.

[6] Berberich, K., Bedathur, S., Alonso, O., and Weikum, G. (2010). A Language Modeling Approach for Temporal Information Needs. LNCS, 5993, 13-25.

[7] Bollegala, D., Matsuo, Y., and Ishizuka, M. (2007). Measuring Semantic Similarity between Words Using Web Search Engines. In *WWW'07*, 757-766. Banff, Canada. May 8-12.

[8] Campos, R. (2011). http://www.ccc.ipt.pt/~ricardo/software

[9] Campos, R., Dias, G., and Jorge, A. M. (2011). What is the Temporal Value of Web Snippets? In *WWW'11-TWAW*, Hyderabad, India.

[10] Campos, R., Jorge, A., & Dias, G. (2011). Using Web Snippets and Query-logs to Measure Implicit Temporal Intents in Queries. In *SIGIR'11-QRU*, 13-16. Beijing, China. July 28.

[11] Church, K., and Hanks, P. (1990). Word Association Norms Mutual Information and Lexicography. In Comp. Linguistics, 16(1), 23-29.

[12] Cilibrasi, R. L., and Vitányi, P. M. (2007). The Google Similarity Distance. In *IEEE TKDE*, 19(3), 370-373

[13] Dakka, W., Gravano, L., and Ipeirotis, P. G. (2008). Answering General Time Sensitive Queries. In CIKM'08, 1437-1438.

[14] Deerwester, S., Dumais, S., Landauer, T., Furnas, G., and Harshman, R. (1990). Indexing by Latent Semantic Analysis. In *Journal of the American Society for Information Science*, 41(6), 391-407.

[15] Dias, G., Alves, E., and Lopes, J. (2007). Topic Segmentation Algorithms for Text Summarization and Passage Retrieval: An Exhaustive Evaluation. In *AAAI'07,* 1334-1340. Canada. July 22-26.

[16] Dice, L. R. (1945). Measures of the Amount of Ecologic Association between Species. In *Ecological Society of America , 26*, 297-302.

[17] Dumais, S. T. (2005). Latent Semantic Analysis. *In Annual Review of Information Science and Technology , 38(1)*, 188-230.

[18] Freitag, D., Blume, M., Byrnes, J., Chow, E., Kapadia, S., Rohwer, R., et al. (2005). New Experiments in Distributional Representations of Synonymy. In *CoNLL'05*, 25-32. Michigan, USA.

[19] Georgetown University. (2002). *GUTime Download*. http://www.timeml.org/site/tarsqi/modules/gutime/download.html

[20] Google Insights (2011). http://www.google.com/insights/search

[21] Ikehara, S., Murakami, J., and Kimoto, Y. (2003). Vector Space Model based on Semantic Attributes of Words. In *JNLP: Journal of Natural Language Processing*, 10(2), 111-128.

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

[23] Jones, R., and Diaz, F. (2007). Temporal Profiles of Queries. In *TOIS: ACM Transactions on Information Systems*, 25(3).

[24] Kanhabua, N., and Nørvåg, K. (2010). Determining Time of Queries for Re-Ranking Search Results. In *ECDL'10*. Glasgow, Scotland.

[25] Katzell, R. A., and Cureton, E. E. (1947). Biserial Correlation and Prediction. In *The Journal of Psychology*, 24(2), 273 - 278.

[26] Machado, D., Barbosa, T., Pais, S., Martins, B., and Dias, G. Universal Mobile Information Retrieval. In *HCII'09*, 345-354. USA.

[27] McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. In *Psychometrika* 12(2), 153 - 157.

[28] Metzler, D., Jones, R., Peng, F., and Zhang, R. (2009). Improving Search Relevance for Implicitly Temporal Queries. In *SIGIR'09*, 700-701. Boston, USA. July 19-23.

[29] Rogers, D. J., and Tanimoto, T. T. (1960). A Computer Program for Classifying Plants. In Science , 132, 1115-1118.

[30] Ruprecht-Karl University Heidelberg. (2011). Temporal Tagging. http://dbs.ifi.uni-heidelberg.de/index.php?id=129#c784

[31] Silva, J. F., Dias, G., Guilloré, S., and Pereira, J. G. (1999). Using LocalMaxs Algorithm for the Extraction of Contiguous and Non-contiguous Multiword Lexical Units. In *EPIA'99*, 21-24. Portugal.

[32] Turney, P. D. (2001). Mining the Web for Synonyms: PMI-IR versus LSA on TOEFL. In *EMCL'01*, 491-502. Freiburg, Germany.