A Comparison of Time-aware Ranking Methods

Nattiya Kanhabua  
Dept. of Computer Science  
Norwegian University of Science and Technology  
Trondheim, Norway  
nattiya@idi.ntnu.no

Kjetil Nørvåg  
Dept. of Computer Science  
Norwegian University of Science and Technology  
Trondheim, Norway  
nervoaga@idi.ntnu.no

ABSTRACT
When searching a temporal document collection, e.g., news archives or blogs, the time dimension must be explicitly incorporated into a retrieval model in order to improve relevance ranking. Previous work has followed one of two main approaches: 1) a mixture model linearly combining textual similarity and temporal similarity, or 2) a probabilistic model generating a query from the textual and temporal part of a document independently. In this paper, we compare the effectiveness of different time-aware ranking methods by using a mixture model applied to all methods. Extensive evaluation is conducted using the New York Times Annotated Corpus, queries and relevance judgments obtained using the Amazon Mechanical Turk.

Categories and Subject Descriptors H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval

General Terms Algorithms, Experimentation, Performance

Keywords Time-aware ranking, Temporal similarity

1. INTRODUCTION
We deal with a retrieval task that a query is explicitly provided with time, i.e., containing temporal information needs. In this case, the time dimension must be incorporated into a retrieval model in order to improve relevance ranking. Consider a query containing the temporal expression “Independence Day 2009”, an existing retrieval model relying on term matching will fail to retrieve a document mentioning “July 4, 2009”, although two temporal expressions refer to the same date. Hence, when dealing with the time dimension, time uncertainty should be taken into account because any two temporal expressions can be relevant even they are not equally written.

The previous time-aware ranking methods [1, 2, 3, 4, 5] are based on two main approaches: 1) a mixture model linearly combining textual and temporal similarity, or 2) a probabilistic model generating a query from the textual and temporal part of a document independently. It is shown that time-aware ranking performs better than keyword-based ranking. To the best of our knowledge, an empirical comparison of different time-aware ranking methods has never been done before. In this paper, we will evaluate the effectiveness of different time-aware ranking methods: LMT [1], LMTU [1], TS [4], TSU [4], and FuzzySet [3] using the same dataset, and we will give a brief discussion of the evaluation.

2. MODEL
A temporal expression or the publication date of a document is represented as a quadruple [1]: \( (tb_l, tb_u, te_l, te_u) \) where \( tb_l \) and \( tb_u \) are the lower bound and upper bound for the begin boundary of a time interval respectively. Similarly, \( te_l \) and \( te_u \) are the lower bound and upper bound for the end boundary of a time interval. A temporal query \( q \) is composed of keywords \( q_{text} \) and temporal expressions \( q_{time} \).

A document \( d \) consists of the textual part \( d_{text} \), i.e., a bag of words, and the temporal part \( d_{time} \) composed of the publication date \( PubTime(d) \), and temporal expressions mentioned in the document’s contents \( ContentTime(d) \) or \( \{t_1, \ldots, t_k\} \).

To be comparable, we apply a mixture model to linearly combine textual similarity and temporal similarity for all ranking methods. Given a temporal query \( q \), a document \( d \) will be ranked according to a score computed as follows:

\[
S(q, d) = (1 - \alpha) \cdot S'(q_{text}, d_{text}) + \alpha \cdot S''(q_{time}, d_{time})
\]

where the mixture parameter \( \alpha \) indicates the importance of textual similarity \( S'(q_{text}, d_{text}) \) and temporal similarity \( S''(q_{time}, d_{time}) \). Both similarity scores must be normalized, e.g., divided by the maximum scores, in order to the final score \( S(q, d) \). \( S'(q_{text}, d_{text}) \) can be measured using any of existing text-based weighting functions. \( S''(q_{time}, d_{time}) \) measure temporal similarity by assuming that a temporal expression \( t_q \in q_{time} \) is generated independently from each other, and a two-step generative model was used [1]:

\[
S''(q_{time}, d_{time}) = \prod_{t_q \in q_{time}} P(t_q|d_{time}) = \prod_{t_q \in q_{time}} \frac{1}{\alpha_{time}} \sum_{t_d \in d_{time}} P(t_q|t_d)
\]

Linear interpolation smoothing will be applied to give the probability \( P(t_q|t_d) \) for an unseen query temporal expression \( t_q \) in \( d \). In the next section, we will explain how to estimate \( P(t_q|t_d) \) for different time-aware ranking methods.

3. TIME-AWARE RANKING METHODS
The time-aware ranking methods we study differ from each other in two main aspects: 1) whether or not time uncertainty is concerned, and 2) whether the publication time or the content time of a document is used in ranking. LMT ignores time uncertainty and it exploits the content time of \( d \). LMT can be calculated as:

\[
P(t_q|t_d)_{LMT} = \begin{cases} 
0 & \text{if } t_q \neq t_d, \\
1 & \text{if } t_q = t_d 
\end{cases}
\]

where \( t_d \in ContentTime(d) \), and the score will be equal to 1 iff a temporal expression \( t_d \) is exactly equal to \( t_q \). LMTU concerns time uncertainty by assuming equal likelihood for any time interval \( t'_q \) that \( t_q \) can refer to, that is, \( t_q = \{t'_q|t'_q \in t_q\} \). The simplified calculation of \( P(t_q|t_d) \) for LMTU is given as:
$P(t_q|t_d)_{LMTU} = \frac{|t_q \cap t_d|}{|t_q| \cdot |t_d|}$

where $t_d \in ContentTime(d)$. The detailed computation of $|t_q \cap t_d|$, $|t_q|$ and $|t_d|$ can be referred to [1].

TS ignores time uncertainty. $P(t_q|t_d)_{TS}$ can be computed similarly to $P(t_q|t_d)_{LMTU}$, but $t_d$ is corresponding to the publication time of $d$ instead of the content time as computed for LMT. TSU exploits the publication time of $d$ as done for TS, but it also takes time-uncertainty into account. $P(t_q|t_d)_{TSU}$ is defined using an exponential decay function:

$$P(t_q|t_d)_{TSU} = \frac{\text{DecayRate}}{\lambda} \frac{t_q - t_d}{\mu}$$

where $t_d = \text{PubTime}(d)$. DecayRate and $\lambda$ are constant, $0 < \text{DecayRate} < 1$ and $\lambda > 0$, and $\mu$ is a unit of time distance. The main idea is to give a score that decreases proportional to the time distance between $t_q$ and $t_d$. The less time distance, the more temporally similar they are.

FuzzySet measures temporal similarity using a fuzzy membership function and it exploits the publication time of $d$ for determining temporal similarity. $P(t_q|t_d)_{FuzzySet}$ is given as:

$$P(t_q|t_d)_{FuzzySet} = \begin{cases} 
0 & \text{if } t_d < a_1, \\
1 & \text{if } t_d \geq a_1 \land t_d \leq a_2, \\
1 & \text{if } t_d > a_2, \\
f_1(t_d) & \text{if } t_d < a_2 \land t_d \leq a_3, \\
f_2(t_d) & \text{if } t_d > a_3 \land t_d \leq a_4, \\
m & \text{if } t_d < a_4, \\
\frac{a_1-t_d}{a_1-a_2} & \text{if } a_1 \neq a_2, \text{ or } 1 \text{ if } a_1 = a_2, \\
\frac{a_3-t_d}{a_3-a_4} & \text{if } a_3 \neq a_4, \text{ or } 1 \text{ if } a_3 = a_4.
\end{cases}$$

where $t_d = \text{PubTime}(d)$. $f_1(t_d)$ is $\left( \frac{a_1-t_d}{a_1-a_2} \right)^n$ if $a_1 \neq a_2$, or $1$ if $a_1 = a_2$. $f_2(t_d)$ is $\left( \frac{a_3-t_d}{a_3-a_4} \right)^n$ if $a_3 \neq a_4$, or $1$ if $a_3 = a_4$. The parameters $a_1, a_2, n, m$ are determined empirically.

### 4. EXPERIMENTS

The New York Times Annotated Corpus is used and 40 queries from [1] obtained using the Amazon Mechanical Turk. Note that, a standard dataset, e.g., TREC, is not applicable because queries are not time-related, and judgments are not targeted towards temporal information needs.

Documents are indexed and retrieved using the Apache Lucene version 2.9.3. There are two modes for retrieval [1]: 1) inclusive and 2) exclusive. For inclusive, both query terms and a temporal expression comprise a query $q_{test}$. For exclusive, only query terms constitute $q_{test}$, and a temporal expression is excluded from $q_{test}$. The baseline is the textual similarity $S^t(q_{test}, a_{test})$, i.e., the Lucene’s default weighting function, using inclusive mode denoted TFIDF-IN.

The smoothing parameter is set to 0.1. Parameters for TSU are DecayRate $= 0.5$, $\lambda = 0.5$, and $\mu = 6$ months. Parameters for FuzzySet are $n = 2$, $m = 2$, $a_1 = a_2 = -0.25 \times (a_3-a_2)$, and $a_3 = a_4 = 0.5 \times (a_3-a_2)$. The effectiveness is measured as the precision at 1, 5 and 10 documents ($P@1$, $P@5$ and $P@10$), mean average precision (MAP), and mean reciprocal rank (MRR). The sensitivity of the effectiveness to the mixture parameter $\alpha$ is depicted in Figure 1. The results show that the effectiveness of LmT and LmTU decreases when $\alpha$ is increased, whereas the effectiveness of all other methods slightly increases with the value of $\alpha$.

Table 1 shows the best performing results of each method. In general, all time-aware ranking methods outperform the baseline significantly, except LMT. For each time-aware ranking, the effectiveness when retrieved using exclusive is better than inclusive. TSU performs best among all methods in both inclusive and exclusive modes, and it outperforms all other methods significantly for $P@1$, MAP and MRR.

### 5. CONCLUSIONS

Time-aware ranking methods show better performance compared to a method based on only keywords. When the time-uncertainty is taken into account, the effectiveness is improved significantly. Even though TSU gains the best performance among other methods, the usefulness of TSU is still limited for a document collection with no time metadata, i.e., the publication time of documents is not available. On the contrary, LmT and LmTU can be applied to any document collection without time metadata, but the extraction of temporal expressions is needed.

### 6. REFERENCES


[4] N. Kanhabua and K. Nørvåg. Determining time of queries for time-uncertainty is taken into account, the effectiveness is improved significantly. Even though TSU gains the best performance among other methods, the usefulness of TSU is still limited for a document collection with no time metadata, i.e., the publication time of documents is not available. On the contrary, LmT and LmTU can be applied to any document collection without time metadata, but the extraction of temporal expressions is needed.

### 6. REFERENCES


