Chapter 8

Comparison of Time-aware Ranking Methods

In general, a time-aware ranking method ranks documents that are textually and temporally similar to a query and ranks retrieved documents with respect to both similarities. 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 chapter, we address the research question: *how to explicitly model the time dimension into retrieval and ranking?*, by performing an empirical study and evaluation of different time-aware ranking methods using the same dataset.

8.1 Motivation

The previous time-aware ranking methods [12, 31, 53, 73] 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, e.g., tf-idf and language modeling. To the best of our knowledge, an empirical comparison of different time-aware ranking methods using the same dataset has never been done before.

Contributions

Our main contributions in this chapter are as follows.

- We perform the first study and analysis of different time-aware ranking methods.
- By conducting extensive experiments, we compare the performance of different time-aware ranking methods using the same dataset.

Organization

The organization of the rest of the chapter is as follows. In Section 8.2, we give an overview of related work. In Section 8.3, we first outline the models for documents and queries, and we present a mixture model of time-aware ranking. In Section 8.4, we describe different time-aware ranking methods, and we conduct extensive experiments in order to evaluate different time-aware ranking methods in Section 8.5. Finally, in Section 8.6, we conclude the chapter.

8.2 Related Work

In this section, we give an overview of ranking methods that incorporate temporal information, and point out their underlying aspects including: 1) explicit or implicit temporal information needs, 2) uncertainty-concern or uncertainty-ignore, and 3) using timestamps or temporal expressions.

A number of ranking models exploiting temporal information have been proposed, including [3, 7, 31, 53, 73, 86]. In [73], Li and Croft incorporated time into language models, called time-based language models, by assigning a document prior using an exponential decay function of the publication time of document, i.e., the creation date. They did not have temporal information needs explicitly provided, but they focused on recency queries. The time uncertainty is captured by the exponential decay function, such that the more recent documents obtain the higher probabilities of relevance.

In [31], Diaz and Jones measure the distribution of creation dates of retrieved documents to create the temporal profile of a query. The temporal profile was presented due to no explicit temporal information needs. Hence, they needed to estimate the time relevant to a query by analyzing the distribution of creation dates. Their results showed that the temporal profile together with the contents of retrieved documents can improve averaged precision for the query by using a set of different features for discriminating between temporal profiles.

In [53], Kalczynski and Chou proposed a temporal retrieval model for news archives. In their work, temporal expressions in a query and documents were explicitly modeled in ranking. A query is defined as a set of precise temporal information needs, i.e., the finest time chronon, or a day. Thus, they assumed that the uncertainty applied only to temporal references in documents, and it was represented as a fuzzy set function.

The work by Baeza-Yates [7] proposed to extract temporal expressions from news, index news articles together with temporal expressions, and retrieve temporal information (in this case, future-related events) by using a probabilistic model. A document score is given by multiplying a *keyword* similarity and a time confidence, i.e., a probability that the document's events will actually happen. We can view the confidence as the uncertainty of time. Besides, this work allowed a user to explicitly specify temporal information needs, but only on a year-level granularity.

Metzler et al. [86] considered implicit temporal information needs. They proposed mining query logs and analyze query frequencies over time in order to identify strongly time-related queries. They did not directly extract temporal expressions from queries and

documents. In addition, they presented a ranking model concerning implicit temporal needs, and the experimental results showed the improvement of the retrieval effectiveness of temporal queries for web search.

In more recent work, Berberich et al. [12] integrated temporal expressions into query-likelihood language modeling, which considers uncertainty inherent to temporal expressions in a query and documents. That is, temporal expressions can refer to the same time interval even they are not exactly equal. The work by Berberich et al. required explicit temporal information needs as a part of query.

We will later detail different time-aware ranking methods: LMT [12], LMTU [12], TS(cf. Chapter 4), TSU(cf. Chapter 4), and FuzzySet [53] that underline different aspects of time and uncertainty in Section 8.4.

8.3 Models for Documents and Queries

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 $\mathit{PubTime}(d)$, and temporal expressions $\{t_1, \ldots, t_k\}$ mentioned in the document's contents $\mathit{ContentTime}(d)$. Both the publication date and temporal expressions will be represented using the time model of Berberich et al. [12] presented in Section 2.2.2.

8.4 Time-aware Ranking Methods

We study different time-aware ranking methods proposed to measure temporal similarity between a query and a document including: LMT [12], LMTU [12], TS (cf. Chapter 4), TSU (cf. Chapter 4), and FuzzySet [53]. Although they are shown the good performance in the retrieval of temporal needs, those methods have never been compared using the same dataset and relevance judgments. In the following, we describe in detail each time-aware ranking method. The summarization of characteristics of the time-aware ranking methods with respect to two aspects is shown in Table 8.1.

Table 8.1: Characteristics of different time-aware ranking models.

Method	Time		Uncertainty	
	Publication	*Content*	*Ignore*	*Concern*
LMT	x	√	√	x
LMTU	x	√	x	√
TS	√	x	√	x
TSU	x	√	x	√
FuzzySet	√	x	x	√

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:

$$S(q,d) = (1-\alpha) \cdot S'(q_{text}, d_{text}) + \alpha \cdot S''(q_{time}, d_{time}) \tag{8.1}$$

where the mixture parameter α 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 [12]:

$$\begin{aligned} S''(q_{time}, d_{time}) &= \prod_{t_q \in q_{time}} P(t_q|d_{time}) \\ &= \prod_{t_q \in q_{time}} \left(\frac{1}{|d_{time}|} \sum_{t_d \in d_{time}} P(t_q|t_d) \right) \end{aligned} \tag{8.2}$$

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.

The temporal ranking methods LMT and LMTU are based on a generative model approach. Similar to a query-likelihood approach, the textual and temporal part of the query q are generated independently from the corresponding parts of the document d as:

$$P(q|d) = P(q_{text}|d_{text}) \times P(q_{time}|d_{time}) \tag{8.3}$$

The textual similarity part $P(q_{text}|d_{text})$ can be determined by an existing text-based query-likelihood approach, e.g., the original Ponte and Croft model [100].

A temporal expression q_{time} are assumed to be generated independently from each other. To generate each temporal expression t_q in q_{time} from d, a two-step generative model was used. First, a document temporal expression t_d is drawn at uniform random from document temporal expressions d_{time}. Second, a query temporal expression t_q in q_{time} is generated from a temporal expression t_d in d.

$$\begin{aligned} P(q_{time}|d_{time}) &= \prod_{t_q \in q_{time}} P(t_q|d_{time}) \\ &= \prod_{t_q \in q_{time}} \left(\frac{1}{|d_{time}|} \sum_{t_d \in d_{time}} P(t_q|t_d) \right) \end{aligned} \tag{8.4}$$

The probability of generating t_q from t_d or $P(t_q|t_d)$ can be calculated using two different methods: LMT and LMTU. The first method ignores the uncertainty, i.e., only temporal

expressions are exactly equal will be considered. Thus, $P(t_q|t_d)$ under LMT can be computed as:

$$P(t_q|t_d)_{LMT} = \begin{cases} 0 & \text{if } t_q \neq t_d, \\ 1 & \text{if } t_q = t_d. \end{cases} \tag{8.5}$$

Contrary to LMT, the ranking method LMTU takes the uncertainty into account, i.e, it assumes equal likelihood for each time interval t'_q that t_q can refer to. More precisely, a set of time intervals $t_q = \{t'_q | t'_q \in t_q\}$ that the user may have had in mind when issuing the query are assumed equally likely. Recall that the number of time intervals in t_q, denoted $|t_q|$, can be very huge. $P(t_q|t_d)$ under LMTU can be calculated as:

$$P(t_q|t_d)_{LMTU} = \frac{1}{|t_q|} \sum_{t'_q \in t_q} P(t'_q|t_d) \tag{8.6}$$

$$P(t'_q|t_d) = \frac{1}{|t_d|} \begin{cases} 0 & \text{if } t'_q \notin t_d, \\ 1 & \text{if } t'_q \in t_d. \end{cases} \tag{8.7}$$

Finally, the simplified calculation of $P(t_q|t_d)$ is given as follows.

$$P(t_q|t_d)_{LMTU} = \frac{|t_q \cap t_d|}{|t_q| \cdot |t_d|} \tag{8.8}$$

As explained in [12], |t| can be computed efficiently for any content time or temporal expression t in two cases as:
(1) if $tb_u \leq te_l$ then |t| can simply be computed as:

$$|t| = (tb_u - tb_l + 1) \cdot (te_u - te_l + 1)$$

(2) if $tb_u > te_l$ then |t| can be computed as:

$$\begin{aligned} |t| &= \sum_{tb=tb_l}^{tb_u} (te_u - max(tb, te_l) + 1) \\ &= (te_l - tb_l + 1) \cdot (te_u - te_l + 1) \\ &+ (tb_u - te_l) \cdot (te_u - te_l + 1) - 0.5 \cdot (tb_u - te_l) \cdot (tb_u - te_l + 1) \end{aligned}$$

Note that, $P(t_q|t_d)$ for both LMT and LMTU methods is prone to the zero-probability problem. Thus, Jelinek-Mercer smoothing is applied, and the estimated value $\hat{P}(t_q|t_d)$ becomes:

$$\hat{P}(t_q|t_d) = (1 - \lambda_1) \cdot \frac{1}{|C_{time}|} \sum_{t_d \in C_{time}} P(t_q|t_d) + \lambda_1 \cdot \frac{1}{|d_{time}|} \sum_{t_d \in d_{time}} P(t_q|t_d) \tag{8.9}$$

where the smoothing parameter $\lambda_1 \in [0, 1]$, and C is the whole document collection.

In Chapter 4, we proposed to measure the temporal similarity using TS and TSU. Instead of using a language modeling approach as in [12], we employed a mixture model approach to combining the time similarity with the textual similarity. The mixture model-based approach is given as:

$$S(q,d) = (1-\alpha)\cdot S'(q_{text}, d_{text}) + \alpha \cdot S''(q_{time}, d_{time}) \tag{8.10}$$

where α is a parameter underlining the importance of both similarity scores: textual similarity $S'(q_{text}, d_{text})$ and temporal similarity $S'(q_{time}, d_{time})$. The textual similarity can be implemented using an existing text-based weighting models, e.g. tf-idf. The value of textual similarity must be normalized using the maximum keyword score among all documents as:

$$S'_{norm}(q_{text}, d_{text}) = \frac{S'(q_{text}, d_{text})}{\max S'(q_{text}, d_{text})} \tag{8.11}$$

$S''(q_{time}, d_{time})$ or the temporal similarity part is defined using two methods: TS and TSU. Both methods ignore temporal expressions in documents, that is, they represented d using the creation date only, and d_{time} is referred to $\textit{PubTime}(d)$.

The probability of generating $q_{time}$from d_{time}, or $S''(q_{time}, d_{time})$ can be computed as:

$$\begin{aligned} S''(q_{time}, d_{time}) &= P(q_{time}|d_{time}) \\ &= \frac{1}{|q_{time}|}\sum_{t_q \in q_{time}} P(t_q|d_{time}) \end{aligned} \tag{8.12}$$

where q_{time} is a set of query temporal expressions. Hence, $P(q_{time}|d_{time})$ is averaged over the probability of generating each temporal expression in q_{time}, or $P(t_q|d_{time})$.

Similar to LMT and LMTU, the probability of generating a time interval t_q given d_{time} (i.e., $\textit{PubTime}(d)$) can be calculated in two ways: 1) ignoring uncertainty, and 2) taking uncertainty into account. By ignoring uncertainty, $P(t_q|d_{time})$ is defined as:

$$P(t_q|d_{time})_{TS} = \begin{cases} 0 & \text{if } \textit{PubTime}(d) \notin t_q, \\ 1 & \text{if } \textit{PubTime}(d) \in t_q. \end{cases} \tag{8.13}$$

In this case, the probability of generating a query temporal expression is equal to 1 only if the publication date of d is in a range of t_q, or it is equal 0 otherwise. In the case where uncertainty is concerned, $P(t_q|d_{time})$ is defined using an exponential decay function:

$$P(t_q|d_{time})_{TSU} = DecayRate^{\lambda_2 \cdot |t_q - t_d|} \tag{8.14}$$

$$|t_q - t_d| = \frac{|tb_l^q - tb_l^d| + |tb_u^q - tb_u^d| + |te_l^q - te_l^d| + |te_u^q - te_u^d|}{4} \tag{8.15}$$

where $t_d = \textit{PubTime}(d)$, $DecayRate$ and λ are constant, $0 < DecayRate < 1$ and $\lambda > 0$, and μ is a unit of time distance. Intuitively, this function gives a probability that

decreases proportional to the difference between a time interval t_q and the publication date of d. A document with its creation date closer to t_q will receive a higher probability than a document with its creation date farther from t_q. Note that, LMTU concerns the uncertainty by exploiting *all possible time interval* inherent in a temporal expression into the calculation, whereas TSU ignore this assumption but TSU concerns the uncertainty by taking account of *a time distance* (i.e., measuring by a decay function) between two time intervals.

The normalization of $S''_{norm}(q_{time}, d_{time})$ can be computed in two ways:

1. uncertainty-ignorant using $P(t_q|d_{time})_{TS}$ defined in Equation 8.13
2. uncertainty-aware using $P(t_q|d_{time})_{TSU}$ defined in Equation 8.14

Finally, the normalized value of $S''_{norm}(q_{time}, d_{time})$ will be substituted $S''(q_{time}, d_{time})$ in Equation 8.10 yielding the normalized score of a document d given a temporal query q with determined time q_{time} as follows:

$$S_{norm}(q, d) = (1 - \alpha) \cdot S'_{norm}(q_{text}, d_{text}) + \alpha \cdot S''_{norm}(q_{time}, d_{time}) \tag{8.16}$$

Kalczynski and Chou [53] measured the temporal similarity between a query and a document using a fuzzy membership function with *t* different shapes, so-called *t*-zoidal. Rather than assuming all time intervals inherent in a temporal expression, they propose to capture the uncertainty of time using the fuzzy membership function. In this work, we only consider the 6-zoidal fuzzy membership function illustrated in Figure 8.1.

The figure depicts a query temporal expression $t_q = [t_a, t_b]$ with the beginning point t_a and the ending point t_b equivalent to the points a_2 and a_3 respectively. The time of document d_{time} can be any point on a timeline. The temporal similarity between q and d will be computed based on the graphical function in this figure. In addition, this method also ignores temporal expressions in documents, i.e., they represented d using the creation date only, and d_{time} is referred to $PubTime(d)$. Thus, *FuzzySet* is defined as:

$$\textit{FuzzySet} = \begin{cases} 0 & \text{if } t_d < a_1, \\ f_1(t_d) & \text{if } t_d \geq a_1 \wedge t_d \leq a_2, \\ 1 & \text{if } t_d > a_2 \wedge t_d \leq a_3, \\ f_2(t_d) & \text{if } t_d > a_3 \wedge t_d \leq a_4, \\ 0 & \text{if } t_d > a_4. \end{cases} \tag{8.17}$$

$$f_1(t_d) = \begin{cases} \left(\frac{a_1 - t_d}{a_1 - a_2}\right)^n & \text{if } a_1 \neq a_2, \\ 1 & \text{if } a_1 = a_2. \end{cases} \tag{8.18}$$

$$f_2(t_d) = \begin{cases} \left(\frac{a_4 - t_d}{a_4 - a_3}\right)^m & \text{if } a_3 \neq a_4, \\ 1 & \text{if } a_3 = a_4. \end{cases} \tag{8.19}$$

where t_d is equal to d_{time}. The parameters a_1, a_4, n, m will be determined empirically.

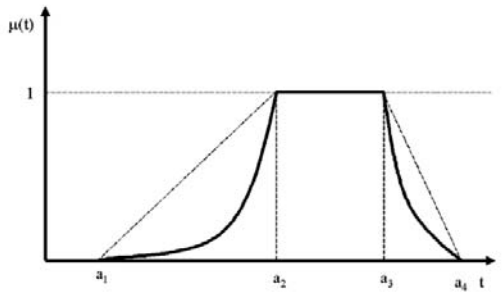

Figure 8.1: The 6-zoidal membership function from [53].

Finally, we will conclude the similarity/dissimilarity of different temporal ranking methods we presents. In other words, we want to remark the difference among them using the following metrics: uncertainty-ignorance or uncertainty-concern, and whether exploiting temporal expressions in either a query or a document, or both.

8.5 Evaluation

We first describe the settings of experiments. Then, we evaluate different time-aware ranking methods, and discuss the results.

8.5.1 Setting

Temporal document collection. We used the New York Times Annotated Corpus as a temporal document collection. Note that, the proposed ranking is not limited this particular collection, but it can be applied to other temporal collections as well. The Apache Lucene search engine version 2.9.3 was used for indexing/retrieving documents.

Queries and relevance assessments. In this work, a standard query and relevance judgment benchmark, such as, TREC, is not useful because queries are not time-related, and the judgment is not targeted towards temporal information needs. For this reason, we used the same set of queries and relevance assessments as the work by Berberich et al [12]. They obtained 40 temporal queries and 6,255 query/document judgments using 5 assessors from the Amazon Mechanical Turk (AMT).

Document annotation. To extract features from the New York Times Annotated Corpus, a series of language processing tools were used as described in [84], including OpenNLP [96] (for tokenization, sentence splitting and part-of-speech tagging, and shallow parsing), the SuperSense tagger [113] (for named entity recognition) and TARSQI Toolkit [118] (for annotating documents with TimeML and extracting temporal expressions). The result of this analysis were: 1) entity information, e.g., all of persons, locations and organizations, 2) temporal expressions, e.g., all of event dates, and 3) sentence information, e.g., all sentences, entities and event dates occurs in each sentence, as well as position information.

Parameter setting. The smoothing parameter was set to 0.1. Parameters for TSU were: $DecayRate = 0.5$, $\lambda = 0.5$, and $\mu = 6$ months. Parameters for FuzzySet were $n = 2$, $m = 2$, $a_1 = a_2 - (0.25 \times (a_3 - a_2))$, and $a_4 = a_3 + (0.50 \times (a_3 - a_2))$.

Evaluating an individual ranking method. To compare different methods, we used a mixture model, where the Lucene's default weighting function was used to capture the textual similarity for all ranking methods. In this way, the results of each temporal ranking can be comparable. The mixture parameter α was varied in the experiments. Each retrieved document is ranked with respect to $S(q, d)$ in Equation 8.16, where $S'(q_{text}, d_{text})$ was a score obtained from the Lucene's default weighting function, and $S''(q_{time}, d_{time})$ was obtained from different time-aware ranking methods described in Section 8.4. The baseline was the textual similarity $S'(q_{text}, d_{text})$, i.e., the Lucene's default weighting function, using *inclusive* mode denoted TFIDF-IN. These two retrieval modes were applied to each temporal ranking method, and the results will be reported accordingly.

Metrics. The retrieval effectiveness of temporal ranking was measured by the precision at 1, 3, 5 and 10 documents (P@1, P@3, P@5 and P@10 respectively), Mean Reciprocal Rank (MRR), and Mean Average Precision (MAP). For the learned ranking method, the average performance over the five folds was used to measure the overall performance of each ranking model.

8.5.2 Results

First, we study the sensitivity of each temporal ranking method to the mixture parameter α. The effectiveness (P@5, P@10, and MAP) of each temporal ranking method when varying α. For *inclusive mode*, the sensitivity of each temporal ranking method is shown in Figure 8.2. For *exclusive mode*, the sensitivity of each temporal ranking method is shown in Figure 8.2. Note that, suffixes IN and EX refer to *inclusive* and *exclusive* mode respectively. Next, we will compare different ranking methods using the best performed results with respect to this sensitivity.

The effectiveness of the baseline (i.e., the Lucene's default weighting function) and different temporal ranking methods are displayed in Table 8.2. In general, the exclusive mode performed better than the inclusive mode for both LMT and LMTU, and LMTU-EX gained the best performance over the other baselines.

Table 8.2 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.

8.6 Conclusions

Time-aware ranking methods show better performance compared to methods based on keywords only. When the time-uncertainty is taken into account, the effectiveness is improved significantly. Even though TSU gains the best performance among other methods,

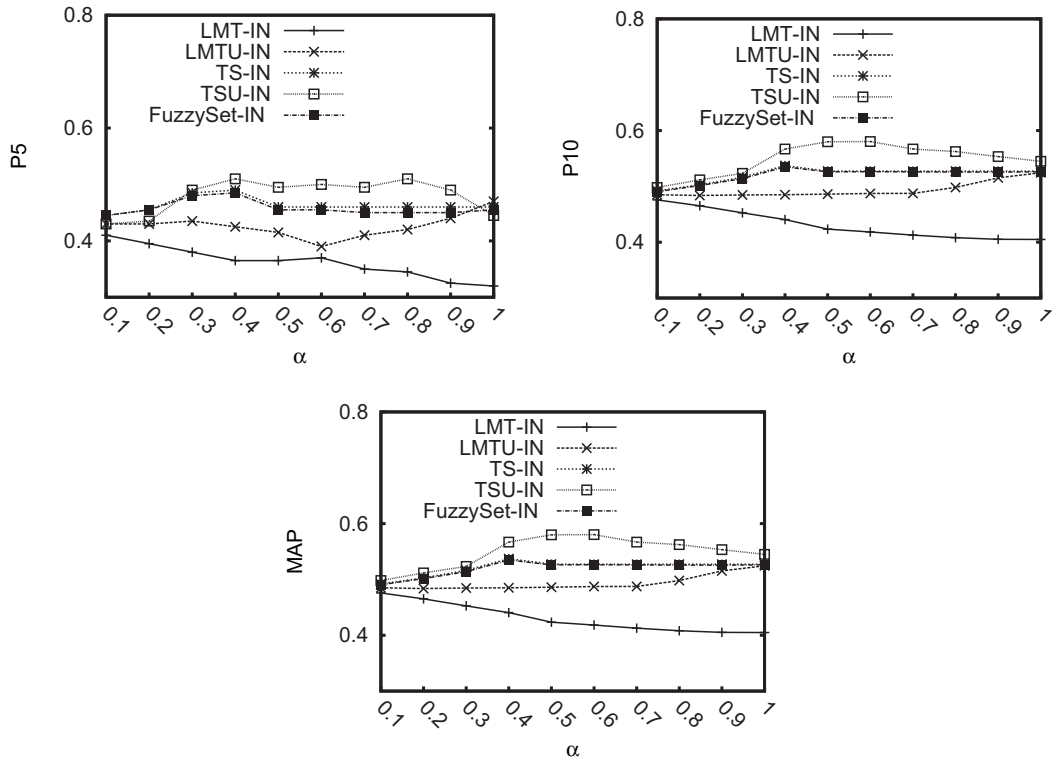

Figure 8.2: Sensitivity of P@5, P@10 and MAP to the mixture parameter α for *inclusive* mode.

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 extraction of temporal expressions is needed.

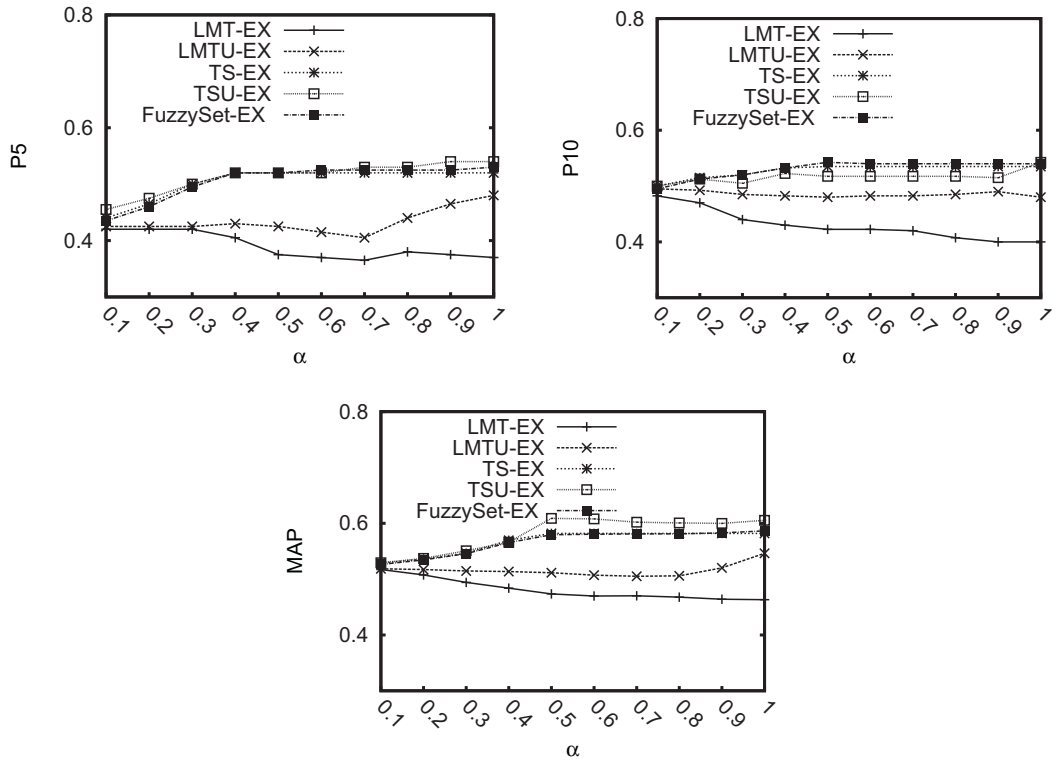

Figure 8.3: Sensitivity of P@5, P@10 and MAP to the mixture parameter α for *exclusive* mode.

Table 8.2: Effectiveness of different time-aware ranking methods (suffixes IN and EX refer to *inclusive* and *exclusive* mode respectively), * indicates statistically improvement over the baselines using t-test with significant at $p < 0.05$.

Methods	P@1	P@3	P@5	P@10	MAP	MRR
TFIDF-IN	.38	.45	.43	.41	.49	.56
LMT-IN	.43	.43	.41	.41	.48	.57
LMTU-IN	.48	.49	.47	.45	.52	.68
TS-IN	.45	.48	.49	.48	.54	.61
TSU-IN	**.65***	.56	.51	.49	**.58***	**.76***
FuzzySet-IN	.45	.48	.49	.48	.53	.61
LMT-EX	.38	.46	.42	.48	.52	.55
LMTU-EX	.48	.52	.48	.50	.55	.68
TS-EX	.48	.56	.52	.53	.58	.63
TSU-EX	**.68***	.58	.54	.54	**.61***	**.77***
FuzzySet-EX	.48	.55	.53	.54	.59	.64