A Language Modeling Approach for Temporal Information Needs

Klaus Berberich, Srikanta Bedathur, Omar Alonso*, and Gerhard Weikum

Max-Planck Institute for Informatics, Saarbrücken, Germany
{kberberi,bedathur,oalonso,weikum}@mpi-inf.mpg.de

Abstract. This work addresses information needs that have a temporal dimension conveyed by a temporal expression in the user's query. Temporal expressions such as "in the 1990s" are frequent, easily extractable, but not leveraged by existing retrieval models. One challenge when dealing with them is their inherent uncertainty. It is often unclear which exact time interval a temporal expression refers to.

We integrate temporal expressions into a language modeling approach, thus making them first-class citizens of the retrieval model and considering their inherent uncertainty. Experiments on the New York Times Annotated Corpus using Amazon Mechanical Turk to collect queries and obtain relevance assessments demonstrate that our approach yields substantial improvements in retrieval effectiveness.

1 Introduction

Many information needs have a temporal dimension as expressed by a temporal phrase contained in the user's query. Existing retrieval models, however, often do not provide satisfying results for such *temporal information needs*, as the following example demonstrates.

Consider a sports journalist, interested in FIFA World Cup tournaments during the 1990s, who issues the query fifa world cup 1990s. A document stating that *France won the FIFA World Cup in 1998* would often not be found by existing retrieval models, despite its obvious relevance to the journalist's information need. The same holds for a document published in 1998 that mentions the *FIFA World Cup final in July*. This is because existing retrieval models miss the semantic connections between the temporal expressions "in 1998" and "in July" contained in the documents and the user's query temporal expression "1990s".

Improving retrieval effectiveness for such temporal information needs is an important objective for several reasons. First, a significant percentage of queries has temporal information needs behind them – about 1.5% of web queries were found to contain an explicit temporal expression (as reported in [1]) and about 7% of web queries have an implicit temporal intent (as reported in [2]). Notice that these numbers are based on general web queries – for *specific domains* (e.g., news or sports) or *expert users* (e.g., journalists or historians) we expect a larger fraction of queries to have a temporal information need behind them. Second,

* Current affiliation: Microsoft Corp.

C. Gurrin et al. (Eds.): ECIR 2010, LNCS 5993, pp. 13–25, 2010.

thanks to improved digitization techniques and preservation efforts, many document collections, including the Web, nowadays contain documents that (i) were *published a long time ago* and (ii) *refer to different times.* Consider, as one such document collection, the archive of the New York Times that covers the years 1851–2009. Articles in this archive provide a contemporary but also retrospective account on events during that time period. When searching these document archives, the temporal dimension plays an important role.

Temporal expressions are frequent across many kinds of documents and can be extracted and resolved with relative ease. However, it is not immediately clear how they should be integrated into a retrieval model. The key problem here is that the actual meaning of many temporal expressions is uncertain, or more specifically, it is not clear which exact time interval they actually refer to. As an illustration, consider the temporal expression "in 1998". Depending on context, it may refer to a particular day in that year, as in the above example about the FIFA World Cup final, or, to the year as a whole as in the sentence *in 1998 Bill Clinton was President of the United States.*

Our approach, in contrast to earlier work [3, 4, 5], considers this uncertainty. It integrates temporal expressions, in a principled manner, into a language modeling approach, thus making them first-class citizens of the retrieval model.

Contributions made in this work are: (i) a novel approach that integrates temporal expressions into a language model retrieval framework and (ii) a comprehensive experimental evaluation on the New York Times Annotated Corpus [6], as a real-world dataset, for which we leverage the crowd-sourcing platform Amazon Mechanical Turk [7] to collect queries and obtain relevance assessments.

Organization. The rest of this paper is organized as follows. In Section 2, we introduce our model and notation. Section 3 describes how temporal expressions can be integrated into a language modeling approach. Conducted experiments and their results are described in Section 4. Section 5 puts our work in context with existing related research. Finally, we conclude in Section 6.

2 Model

In this work, we apply a discrete notion of time and assume the integers $\mathbb{Z}$ as our *time domain* with timestamps $t \in \mathbb{Z}$ denoting the number of time units (e.g., milliseconds or days) passed (to pass) since (until) a reference time-point (e.g., the UNIX epoch). These time units will be referred to as *chronons* in the remainder. We model a temporal expression T as a quadruple

$$T = (\, tb_l,\, tb_u,\, te_l,\, te_u \,) \,.$$

In our representation tb_l and tb_u are respectively a lower bound and upper bound for the begin boundary of the time interval – marking the time interval's earliest and latest possible begin time. Analogously, te_l and te_u are respectively a lower bound and upper bound for the end boundary of the time interval – marking the time interval's earliest and latest possible end time. Since the time interval is not necessarily known exactly, we hence capture lower and upper bounds for its boundaries. To give a concrete example, the temporal expression "in 1998" from the introduction is represented as

$$(\,\texttt{1998/01/01},\ \texttt{1998/12/31},\ \texttt{1998/01/01},\ \texttt{1998/12/31}\,)\ .$$

This representation thus captures the uncertainty inherent to many temporal expressions – a temporal expression T can refer to any time interval $[b,\, e]$ having a begin point $b \in [\, tb_l,\, tb_u\,]$ and an end point $e \in [\, te_l,\, te_u\,]$ along with the constraint $b \leq e$. We consider these time intervals thus as our *elementary units of meaning* in this work. In the remainder, when we refer to the temporal expression T, we implicitly denote the set of time intervals that T can refer to. Note that for notational convenience we use the format YYYY/MM/DD to represent chronons – their actual values are integers as described above.

Let D denote our document collection. A document $d \in D$ is composed of its *textual part* d_{text} and its *temporal part* d_{time}. The textual part d_{text} is a bag of textual terms drawn from a vocabulary V. The temporal part d_{time} is a bag of temporal expressions.

Analogously, a query q also consists of a textual part q_{text} and a temporal part q_{time}. We distinguish two modes how we derive such a query from the user's input, which differ in how they treat temporal expressions extracted from the input. In the *inclusive* mode, the parts of the user's input that constitute a temporal expression are still included in the textual part of the query. In the *exclusive* mode, these are no longer included in the textual part. Thus, for the user input boston july 4 2002, as a concrete example, in the inclusive mode we obtain $q_{text} = \{\mathsf{boston}, \mathsf{july}, \mathsf{4}, \mathsf{2002}\}$, whereas we obtain $q_{text} = \{\mathsf{boston}\}$ in the exclusive mode.

3 Language Models Integrating Temporal Expressions

With our formal model and notation established, we now turn our attention to how temporal expressions can be integrated into a language modeling approach and how we can leverage them to improve retrieval effectiveness for temporal information needs.

We use a query-likelihood approach and thus rank documents according to their estimated probability of generating the query. We assume that the textual and temporal part of the query q are generated independently from the corresponding parts of the document d, yielding

$$P(\,q\,|\,d\,) = P(\,q_{text}\,|\,d_{text}\,) \times P(\,q_{time}\,|\,d_{time}\,)\ . \tag{1}$$

The first factor $P(\,q_{text}\,|\,d_{text}\,)$ can be implemented using an existing text-based query-likelihood approach, e.g., the original Ponte and Croft model [8]. In our concrete implementation, as detailed in Section 4, we employ a unigram language model with Jelinek-Mercer smoothing as described in Manning et al. [9].

For the second factor in (1), we assume that query temporal expressions in q_{time} are generated independently from each other, i.e.,

$$P(\,q_{time}\,|\,d_{time}\,) = \prod_{Q\,\in\,q_{time}} P(\,Q\,|\,d_{time})\ . \tag{2}$$

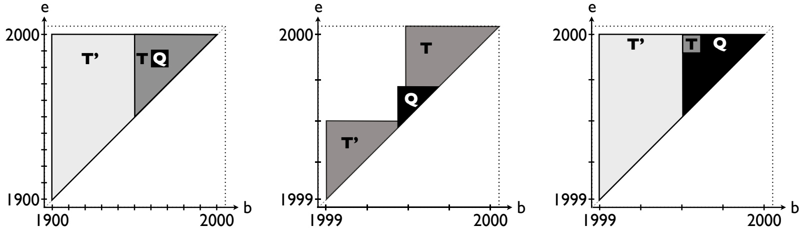
For the generation of temporal expressions from a document d we use a two-step generative model. In the first step, a temporal expression T is drawn at uniform random from the temporal expressions contained in the document. In the second step, a temporal expression is generated from the temporal expression T just drawn. Under this model, the probability of generating the query temporal expression Q from document d is

$$P(\,Q \mid d_{\,time}) = \frac{1}{|d_{\,time}|} \sum_{T \in d_{\,time}} P(\,Q \mid T) \;. \tag{3}$$

In the remainder of this section we describe two ways how $P(\,Q \mid T\,)$ can be defined. Like other language modeling approaches, our model is prone to the zero-probability problem – if one of the query temporal expressions has zero probability of being generated from the document, the probability of generating the query from this document is zero. To mitigate this problem, we employ Jelinek-Mercer smoothing, and estimate the probability of generating the query temporal expression Q from document d as

$$P(\,Q|d_{\,time}) = (1-\lambda)\cdot\frac{1}{|D_{\,time}|} \sum_{T \in D_{\,time}} P(\,Q|T)+\lambda\cdot\frac{1}{|d_{\,time}|} \sum_{T \in d_{\,time}} P(\,Q|T) \tag{4}$$

where $\lambda \in [0,1]$ is a tunable mixing parameter, and $D_{\,time}$ refers to the temporal part of the document collection treated as a single document.



(a) Specificity (b) Coverage (c) Maximality

Fig. 1. Three requirements for a generative model

Before giving two possible definitions of $P(\,Q \mid T\,)$, we identify the following requirements that any definition of $P(\,Q \mid T\,)$ must satisfy (these are illustrated in Figure 1, where each temporal expression is represented as a two-dimensional region that encompasses compatible combinations of begin point b and end point e):

- **Specificity:** Given two temporal expressions T and T', we have

$$|T \cap Q| = |T' \cap Q| \wedge |T| \leq |T'| \Rightarrow P(\,Q \mid T\,) \geq P(\,Q \mid T'\,) \;.$$

In other words, a query temporal expression is more likely to be generated from a temporal expression that closely matches it. Referring to Figure 1(a), the probability of generating Q (corresponding, e.g., to "from the 1960s until the 1980s") from T (corresponding, e.g., to "in the second half of the 20th century") is more than generating it from T' (corresponding, e.g., to "in the 20th century").

- **Coverage:** Given two temporal expressions T and T', we have

$$|T| = |T'| \wedge |T \cap Q| \leq |T' \cap Q| \Rightarrow P(\, Q \mid T\,) \leq P(\, Q \mid T'\,) \; .$$

In this requirement, we capture the notion that a larger overlap with the query temporal expression is preferred. In Figure 1(b), the overlap of Q (corresponding, e.g., to "in the summer of 1999") with T (corresponding, e.g., to "in the first half of 1999") is more than the overlap with T' (corresponding, e.g., to "in the second half of 1999"). Therefore, the latter temporal expression is preferable and should have a higher probability of generating Q.

- **Maximality:** $P(\, Q \mid T\,)$ should be maximal for $T = Q$, i.e.,

$$T \neq Q \Rightarrow P(\, Q \mid T\,) \leq P(\, Q \mid Q\,) \; .$$

This requirement captures the intuition that the probability of generating a query temporal expression from a temporal expression matching it exactly must be the highest. As shown in Figure 1(c), the probability of generating Q (corresponding, e.g., to "in the second half of 1999") from itself should be higher than the probability of generating it from T (corresponding, e.g., to "from July 1999 until December 1999") or T' (corresponding, e.g., to "in 1999"). Note that the maximality requirement can be derived by combining the requirements of specificity and coverage.

3.1 Uncertainty-Ignorant Language Model

Our first approach, referred to as LmT in the remainder, ignores the uncertainty inherent to temporal expressions. A temporal expression T can only generate itself, i.e.,

$$P(\, Q \mid T) = \mathbb{1}(T = Q) \; , \tag{5}$$

where $\mathbb{1}(T = Q)$ is an indicator function whose value assumes 1 iff $T = Q$ (i.e., $tb_l = qb_l \wedge tb_u = qb_u \wedge te_l = qe_l \wedge te_u = qe_u$). The approach thus ignores uncertainty, since it misses the fact that a temporal expression T and a query temporal expression Q may refer to the same time interval, although $T \neq Q$. It can easily be verified that this approach meets the above requirements[2].

Despite its simplicity the approach still profits from the extraction of temporal expressions. To illustrate this, consider the two temporal expressions "in the 1980s" and "in the '80s". Both share the same formal representation in our model, so that LmT can generate a query containing one of them from a document containing the other. In contrast, a text-based approach (i.e., one not paying special attention to temporal expressions), would not be aware of the semantic connection between the textual terms '80s and 1980s.

[2] Proofs and additional details are provided in our accompanying technical report [10].

3.2 Uncertainty-Aware Language Model

As explained in the introduction, for many temporal expressions the exact time interval that they refer to is uncertain. Our second approach LmtU explicitly considers this uncertainty. In detail, we define the probability of generating Q from the document d as

$$P(\,Q \mid T\,) = \frac{1}{|Q|} \sum_{[q_b,\, q_e] \in Q} P(\,[q_b,\, q_e] \mid T\,)\;, \tag{6}$$

where the sum ranges over all time intervals included in Q. The approach thus assumes equal likelihood for each time interval $[q_b,\, q_e]$ that Q can refer to. Intuitively, each time interval that the user may have had in mind when uttering Q is assumed equally likely. Recall that $|Q|$ denotes the huge but finite total number of such time intervals.

The probability of generating the time interval $[q_b,\, q_e]$ from a temporal expression T is defined as

$$P(\,[q_b,\, q_e] \mid T\,) = \frac{1}{|T|}\, \mathbb{1}(\,[q_b,\, q_e] \in T\,) \tag{7}$$

where $\mathbb{1}(\,[q_b,\, q_e] \in T\,)$ is an indicator function whose value is 1 iff $[q_b,\, q_e] \in T$. For T we thus also assume all time intervals that it can refer to as equally likely. Putting (6) and (7) together we obtain

$$P(\,Q \mid T\,) = \frac{1}{|Q|} \sum_{[q_b,\, q_e] \in Q} \frac{1}{|T|}\, \mathbb{1}(\,[q_b,\, q_e] \in T\,)\;, \tag{8}$$

which can be simplified as

$$P(\,Q \mid T\,) = \frac{|T \cap Q|}{|T| \cdot |Q|}\;. \tag{9}$$

Both Q and T are inherently uncertain, i.e., it is not clear which time interval the user issuing the query and author writing the document had in mind when uttering Q and T, respectively. Having no further information, our model assumes equal likelihood for all possible time intervals that Q and T respectively can refer to. This definition meets our three requirements defined above[2].

Efficient Computation. For the practical applicability of this model, one important issue that needs addressing is the efficient evaluation of (9). Naïvely enumerating all time intervals that T and Q can refer to, before computing $|T \cap Q|$ is clearly not a practical solution. Consider again the temporal expression (`1998/01/01`, `1998/12/31`, `1998/01/01`, `1998/12/31`). For a temporal resolution with chronons corresponding to days (hours) the total number of time intervals that this temporal expression can refer to is $66,795$ $(38,373,180)$. Fortunately, though, there is a more efficient way to evaluate (9).

Given $T = (\,tb_l,\, tb_u,\, te_l,\, te_u\,)$, we can compute $|T|$ by distinguishing two cases: (i) if $tb_u \leq te_l$ then $|T|$ can simply be computed as

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

since any begin point b is compatible with any end point e, (ii) otherwise, if $tb_u > te_l$, we can compute $|T|$ as

$$\begin{aligned} |T| &= \sum_{b=tb_l}^{tb_u} (te_u - max(b,\, te_l) + 1) \\ &= (te_l - tb_l + 1) \cdot (te_u - te_l + 1) + (tb_u - te_l) \cdot (te_u - te_l + 1) \\ &\quad - 0.5 \cdot (tb_u - te_l) \cdot (tb_u - te_l + 1) \; . \end{aligned} \tag{10}$$

This captures that only end points $e \geq max(b,\, te_l)$ are compatible with a fixed begin point b. The derivation of (10) is given in the appendix.

Let $Q = (\, qb_l,\, qb_u,\, qe_l,\, qe_u \,)$ be a query temporal expression. Using (10) we can determine $|Q|$. For computing $|Q \cap T|$ notice that each time interval $[b,\, e] \in Q \cap T$ fulfills $b \in [tb_l,\, tb_u] \cap [qb_l,\, qb_u]$ and $e \in [te_l,\, te_u] \cap [qe_l,\, qe_u]$. Therefore, $|T \cap Q|$ can be computed by considering the temporal expression

$$(\, max(tb_l, qb_l),\, min(tb_u, qb_u),\, max(te_l, qe_l),\, min(te_u, qe_u) \,) \; .$$

Thus, we have shown that the generative model underlying LmtU allows for efficient computation. When processing a query with a query temporal expression Q, we need to examine all temporal expressions T with $T \cap Q \neq \emptyset$ and the documents that contain them, as can be seen from (9). This can be implemented efficiently by keeping a small inverted index in main memory that keeps track of the documents that contain a specific temporal expression. Its lexicon, which consists of temporal expressions, can be organized, for instance, using interval trees to support the efficient identification of qualifying temporal expressions via interval intersection.

4 Experimental Evaluation

This section describes the experimental evaluation of our approach.

4.1 Setup and Datasets

Methods under Comparison in our experimental evaluation are:

- Lm(γ) – Unigram language model with Jelinek-Mercer smoothing
- LmT-IN(γ, λ) – Uncertainty-ignorant method using inclusive mode
- LmT-EX(γ, λ) – Uncertainty-ignorant method using exclusive mode
- LmtU-IN(γ, λ) – Uncertainty-aware method using inclusive mode
- LmtU-EX(γ, λ) – Uncertainty-aware method using exclusive mode

Apart from our baseline Lm, we thus consider all four combinations of (a) inclusive vs. exclusive mode (i.e., whether query terms constituting a temporal expression are part of q_{text}) and (b) uncertainty-ignorant vs. uncertainty-aware definition of $P(\, Q | T \,)$. The mixing parameters γ and λ control the Jelinek-Mercer smoothing used when generating the textual part and the temporal part of the query, respectively. We consider values in $\{0.25,\, 0.5,\, 0.75\}$ for each of them, giving us a total of 39 method configurations under comparison. Further, notice

	Sports	**Culture**
Day	boston red sox [october 27, 2004] ac milan [may 23, 2007]	kurt cobain [april 5, 1994] keith harring [february 16, 1990]
Month	stefan edberg [july 1990] italian national soccer team [july 2006]	woodstock [august 1994] pink floyd [march 1973]
Year	babe ruth [1921] chicago bulls [1991]	rocky horror picture show [1975] michael jackson [1982]
Decade	michael jordan [1990s] new york yankees [1910s]	sound of music [1960s] mickey mouse [1930s]
Century	la lakers [21st century] soccer [21st century]	academy award [21st century] jazz music [21st century]
	Technology	**World Affairs**
Day	mac os x [march 24, 2001] voyager [september 5, 1977]	berlin [october 27, 1961] george bush [january 18, 2001]
Month	thomas edison [december 1891] microsoft halo [june 2000]	poland [december 1970] pearl harbor [december 1941]
Year	roentgen [1895] wright brothers [1905]	nixon [1970s] iraq [2001]
Decade	internet [1990s] sewing machine [1850s]	vietnam [1960s] monica lewinsky [1990s]
Century	musket [16th century] siemens [19th century]	queen victoria [19th century] muhammed [7th century]

Fig. 2. Queries categorized according to their topic and temporal granularity

that our baseline Lm, not aware of temporal expressions, always only considers q_{text} as determined using the inclusive mode.

Document Collection. As a dataset for our experimental evaluation we use the publicly-available *New York Times Annotated Corpus* [6] that contains about 1.8 million articles published in New York Times (NYT) between 1987 and 2007.

Queries. Since we target a specific class of information needs, query workloads used in benchmarks like TREC [11] are not useful in our setting. To assemble a query workload that captures users' interests and preferences, we ran two user studies on Amazon Mechanical Turk (AMT). In our first study, workers were provided with an entity related to one of the topics *Sports*, *Culture*, *Technology*, or *World Affairs* and asked to specify a temporal expression that fits the given entity. In our second study, users were shown a temporal expression corresponding to a *Day*, *Month*, *Year*, *Decade*, or *Century* and asked to add an entity related to one of the aforementioned topics. Among the queries obtained from our user studies, we selected the 40 queries shown in Figure 2. Queries are categorized according to their topic and temporal granularity, giving us a total of 20 query categories, each of which contains two queries.

Relevance Assessments were also collected using AMT. We computed top-10 query results for each query and each method configuration under comparison, pooled them, yielding a total of $1,251$ query-document pairs. Each of these query-document pairs was assessed by five workers on AMT. Workers could state whether they considered the document *relevant* or *not relevant* to the query. To prevent spurious assessments, a third option (coined *I don't know*) was provided, which workers should select if they had insufficient information or knowledge to assess the document's relevance. Further, we asked workers to explain in their own words, why the document was relevant or not relevant. We found the feedback provided through the explanations extremely insightful. Examples of provided explanations are:

- roentgen [1895]: *Wilhelm Roentgen was alive in 1895 when the building in New York at 150 Nassau Street in downtown Manhattan, NYC was built, they do not ever intersect other than sharing the same timeline of existence for a short while.*
- nixon [1970s]: *This article is relevant. It is a letter to the editor in response to a column about 1970s-era Nixon drug policy.*
- keith harring [february 16, 1990]: *The article does not have any information on Keith Harring, only Laura Harring. Though it contains the keywords Harring and 1990, the article is obviously not what the searcher is looking for.*

Apart from that, when having to explain their assessment, workers seemed more thorough in their assessments. Per completely assessed query-document pair we paid \$0.02 per assignment to workers. Workers chose relevant for 33%, not relevant for 63%, and the third option (i.e., *I don't know*) for 4% of the total $6,255$ relevance assessments. Relevance assessments with the last option are ignored when computing retrieval-effectiveness measures below.

Implementation Details. We implemented all methods in Java. All data was kept in an Oracle 11g database. Temporal expressions were extracted using TARSQI [12]. TARSQI detects and resolves temporal expressions using a combination of hand-crafted rules and machine learning. It annotates a given input document using the TimeML [13] markup language. Building on TARSQI's output, we extracted range temporal expressions such as "from 1999 until 2002", which TARSQI does not yet support. Further, we added each article's publication date as an additional temporal expression. We map temporal expressions to our quadruple representation using milliseconds as chronons and the UNIX epoch as our reference time-point.

4.2 Experimental Results

We measure the retrieval effectiveness of the methods under comparison using Precision at k (P@k) and nDCG at k (N@k) as two standard measures. When computing P@k, we employ majority voting. Thus, a document is considered relevant to a query, if the majority of workers assessed it as relevant. When computing N@k, the average relevance grade assigned by workers is determined interpreting *relevant* as grade 1 and *not relevant* as grade 0, respectively.

Table 1. Retrieval effectiveness overall

	P@5	**N@5**	**P@10**	**N@10**
Lm ($\gamma = 0.25$)	0.33	0.34	0.30	0.32
Lm ($\gamma = 0.75$)	0.38	0.39	0.37	0.38
LmT-IN ($\gamma = 0.25, \lambda = 0.75$)	0.26	0.27	0.23	0.25
LmT-IN ($\gamma = 0.75, \lambda = 0.75$)	0.29	0.31	0.25	0.28
LmT-EX ($\gamma = 0.25, \lambda = 0.75$)	0.36	0.36	0.32	0.33
LmT-EX ($\gamma = 0.5, \lambda = 0.75$)	0.37	0.37	0.32	0.33
LmtU-IN ($\gamma = 0.25, \lambda = 0.75$)	0.41	0.42	0.37	0.37
LmtU-IN ($\gamma = 0.75, \lambda = 0.25$)	0.44	0.44	0.39	0.40
LmtU-EX ($\gamma = 0.25, \lambda = 0.75$)	0.53	0.51	0.49	0.49
LmtU-EX ($\gamma = 0.5, \lambda = 0.75$)	**0.54**	**0.52**	**0.51**	**0.49**

Overall. Table 1 gives retrieval-effectiveness figures computed using all queries and cut-off levels $k = 5$ and $k = 10$. For each of the five methods under comparison, the table shows the best-performing and worst-performing configuration with their respective parameter values γ and λ. The figures shown support the following three observations: (i) the exclusive mode outperforms the inclusive mode for both LmT and LmtU, (ii) LmT does not yield improvements over the baseline Lm, but (iii) LmtU is at par with the baseline Lm when the inclusive mode is used and outperforms it significantly when used with the exclusive mode. For LmtU-EX the worst configuration beats the best configuration of the baseline. Further, the worst and best configuration of LmtU-EX are close to each other demonstrating the method's robustness.

Table 2. Retrieval effectiveness by topic

	Sports		Culture		Technology		World Affairs	
	P@10	N@10	P@10	N@10	P@10	N@10	P@10	N@10
Lm	0.33	0.33	0.39	0.38	0.27	0.32	0.50	0.49
LmT-IN	0.36	0.36	0.25	0.30	0.10	0.15	0.30	0.30
LmT-EX	0.46	0.44	0.33	0.34	0.12	0.17	0.38	0.38
LmtU-IN	0.46	0.44	0.41	0.42	0.21	0.27	0.48	0.48
LmtU-EX	**0.67**	**0.58**	**0.47**	**0.49**	**0.29**	**0.34**	**0.60**	**0.57**

By Topic. For the best-performing configuration of each method (as given in Table 1), we compute retrieval-effectiveness measures at cut-off level $k = 10$ and group them by topic. Table 2 shows the resulting figures. These support our above observations. In addition, we observe that all methods perform worst on queries from *Technology*. The best performance varies per method and measure.

Table 3. Retrieval effectiveness by temporal granularity

	Day		Month		Year		Decade		Century	
	P@10	N@10	P@10	N@10	P@10	N@10	P@10	N@10	P@10	N@10
Lm	0.35	0.38	0.42	0.40	0.65	0.59	0.20	0.28	0.25	0.26
LmT-IN	0.18	0.22	0.20	0.21	0.55	0.50	0.23	0.30	0.20	0.24
LmT-EX	0.26	0.28	0.24	0.25	0.58	0.55	0.28	0.33	0.31	0.32
LmtU-IN	0.33	0.36	0.47	0.46	0.59	0.56	0.34	0.35	0.24	0.27
LmtU-EX	**0.43**	**0.44**	**0.50**	**0.50**	**0.69**	**0.64**	**0.56**	**0.54**	**0.36**	**0.35**

By Temporal Granularity. In analogy, we can group retrieval-effectiveness measurements at cut-off level $k = 10$ by temporal granularity – again considering only the best-performing configuration of each method. Table 3 gives the resulting figures. Again, LmtU-EX consistently achieves the best retrieval performance. We further observe significant variations in retrieval effectiveness across temporal granularities. For queries including a year, all methods achieve their best performance. The worst performance varies per method and measure.

Summary. Putting things together, there is a clear winner in our experimental evaluation. LmtU-EX consistently achieves the best retrieval performance. This demonstrates that (i) considering the uncertainty inherent to temporal expressions is essential and (ii) excluding terms that constitute a temporal expression from the textual part of the query is beneficial. These findings are confirmed by a second experiment on a snapshot of the English Wikipedia [14][2].

5 Related Work

We now put our work in context with existing prior research. Alonso et al. [15] highlight the importance of temporal information in Information Retrieval, and suggest the problem addressed in this work as one not yet satisfactorily supported by existing approaches.

Li and Croft [16] and Dakka et al. [17] both propose language models that take into account publication times of documents, in order to favor, for instance, more recent documents. Kanahuba and Nørvåg [18] and de Jong et al. [19] employ language models to date documents, i.e., determine their publication time. Del Corso et al. [20] address the problem of ranking news articles, taking into account publication times but also their interlinkage. Jones and Diaz [21] focus on constructing query-specific temporal profiles based on the publication times of relevant documents. Thus, all of the approaches mentioned are based on the publication times of documents. None of the approaches, though, considers temporal expressions contained in the documents' contents.

Baeza-Yates [4] is the earliest approach that considers temporal expressions contained in documents for retrieval purposes. It aims at searching information that refers to the future. The proposed retrieval model is focused on confidences associated with statements about the future, thus favoring relevant documents that are confident about their predictions regarding a future time of interest. Kalczynski et al. [5] study the human perception of temporal expressions and propose a retrieval model for business news archives that takes into account temporal expressions. Arikan et al. [3] integrate temporal expressions into a language modeling approach but ignore the aspect of uncertainty. Metzler et al. [2], most recently, identify so-called implicitly temporal queries and propose a method to bias ranking functions in favor of documents matching the user's implicit temporal intent.

The extraction of temporal expressions is a well-studied problem. For an overview of the current state of the art and a description of the TARSQI toolkit, we refer to Verhagen et al. [12, 22]. Our formal representation of temporal expressions as quadruples is adopted from Zhang et al. [23]. Koen and Bender [24] describe the Time Frames system that extracts temporal expressions and uses them to augment the user experience when reading news articles, for instance, by displaying a temporal context of concurrent events.

Several prototypes are available that make use of temporal expressions when searching the Web, most notably, Google's Timeline View [25] and TimeSearch [26]. Details about their internals, though, have not been published.

Crowd-sourcing platforms such as AMT are becoming a common tool for conducting experiments in Information Retrieval. For a discussion of their benefits and guidelines on how to use them, we refer to Alonso et al. [27].

6 Conclusion

In this work, we have developed a novel approach that integrates temporal expressions into a language model retrieval framework, taking into account the uncertainty inherent to temporal expressions. Comprehensive experiments on a large corpus of New York Times articles with relevance assessments obtained using Amazon Mechanical Turk showed that our approach substantially improves retrieval effectiveness for temporal information needs.

Ongoing and Future Work. Our focus in this work has been on temporal information needs disclosed by an *explicit* temporal expression in the user's query. Often, as somewhat explored in [2], queries may not contain such an explicit temporal expression, but still have an associated *implicit* temporal intent. Consider a query such as bill clinton arkansas that is likely to allude to Bill Clinton's time as Governor of Arkansas between 1971 and 1981. Detecting and dealing with such queries is part of our ongoing research.

Acknowledgment

This work was partially supported by the EU within the 7th Framework Programme under contract 216267 "Living Web Archives (LiWA)".

References

[1] Nunes, S., et al.: Use of Temporal Expressions in Web Search. In: ECIR (2008)
[2] Metzler, D., et al.: Improving Search Relevance for Implicitly Temporal Queries. In: SIGIR (2009)
[3] Arikan, I., et al.: Time Will Tell: Leveraging Temporal Expressions in IR. In: WSDM (2009)
[4] Baeza-Yates, R.A.: Searching the future. In: SIGIR Workshop MF/IR (2005)
[5] Kalczynski, P.J., Chou, A.: Temporal document retrieval model for business news archives. Inf. Process. Manage. (2005)
[6] New York Times Annotated Corpus, `http://corpus.nytimes.com`
[7] Amazon Mechanical Turk, `http://www.mturk.com`
[8] Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR (1998)
[9] Manning, C.D., et al.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
[10] Berberich, K., et al.: A Language Modeling Approach for Temporal Information Needs. Research Report MPI-I-2010-5-001
[11] Text REtrievel Conference, `http://trec.nist.gov`
[12] Verhagen, M., et al.: Automating Temporal Annotation with TARSQI. In: ACL (2005)
[13] TimeML Specification Language, `http://www.timeml.org`
[14] Wikipedia, `http://www.wikipedia.org`
[15] Alonso, O., et al.: On the value of temporal information in information retrieval. SIGIR Forum (2007)
[16] Li, X., Croft, W.B.: Time-based language models. In: CIKM (2003)
[17] Dakka, W., et al.: Answering general time sensitive queries. In: CIKM (2008)

[18] Kanhabua, N., Nørvåg, K.: Improving temporal language models for determining time of non-timestamped documents. In: ECDL (2008)

[19] de Jong, F. et al.: Temporal language models for the disclosure of historical text. In: AHC (2005)

[20] Corso, G.M.D., et al.: Ranking a stream of news. In: WWW (2005)

[21] Jones, R., Diaz, F.: Temporal profiles of queries. ACM Trans. Inf. Syst. (2007)

[22] Verhagen, M., Moszkowicz, J.L.: Temporal Annotation and Representation. In: Language and Linguistics Compass (2009)

[23] Zhang, Q., et al.: TOB: Timely Ontologies for Business Relations. In: WebDB (2008)

[24] Koen, D.B., Bender, W.: Time frames: Temporal augmentation of the news. IBM Systems Journal (2000)

[25] Google's Timeline View, `http://www.google.com/experimental/`

[26] TimeSearch History, `http://www.timesearch.info`

[27] Alonso, O., et al.: Crowdsourcing for relevance evaluation. SIGIR Forum (2008)

Appendix: Derivation of Equation 10

Recall that we assume $tb_u > te_l$.

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