Identifying Time Intervals of Interest to Queries

Dhruv Gupta[†] [*] Klaus Berberich[†]

[†]Max Planck Institute for Informatics
Saarbrücken, Germany
{dhgupta,kberberi}@mpi-inf.mpg.de

[*]IIT Patna
Patna, India
dhruv.mc12@iitp.ac.in

ABSTRACT

We investigate how time intervals of interest to a query can be identified automatically based on pseudo-relevant documents, taking into account both their publication dates and temporal expressions from their contents. Our approach is based on a generative model and is able to determine time intervals at different temporal granularities (e.g., day, month, or year). We evaluate our approach on twenty years' worth of newspaper articles from The New York Times using two novel testbeds consisting of temporally unambiguous and temporally ambiguous queries, respectively.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]

Keywords

Temporal Information Retrieval

1. INTRODUCTION

Time has been recognized as an important dimension in Information Retrieval [2], and recent years have seen an increased interest in making use of temporal information associated with documents or information needs. Tasks that have been tackled include retrieving recent relevant documents [10] as well as documents relevant to implicitly [11] or explicitly [3, 4] temporal queries. Beyond that, also web search engines have meanwhile deployed features to keep up with the changing Web, indexing recently published documents, and filter results based on their publication dates.

In this work, we address the problem of automatically identifying time intervals of interest to a given keyword query. For instance, when presented with the keyword query bill clinton presidency, a good time interval to determine would be [1993, 2001], which covers the years of Clinton's presidency. This is a useful building block in temporal information retrieval with applications such as (i) temporal query reformulation and expansion – by adding time intervals of interest to the query, (ii) temporal diversification of search results – by making sure that the result covers diverse time intervals of interest to the query, and (iii) providing more structured query results to users – organized by important time intervals they refer to.

While ours is not the first effort in this direction, it differs from previous ones [4, 9] in several important aspects. First, our focus is on time intervals (e.g., [1993, 2001]) as opposed to time points at a fixed temporal granularity (e.g., the years 1993 and 2001). Second, we make use of both documents' publication dates, as part of their meta data, as well as temporal expressions from their contents. Third, our approach is not restricted to a fixed temporal granularity but can determine time intervals of interest at different temporal granularities (e.g., day, month, and year). Finally, we also consider temporally ambiguous queries for which more than one time interval is of interest – say george bush presidency or san francisco earthquake.

This work builds on prior research [3], which aims at improving retrieval effectiveness for explicitly temporal queries such as summer olympics 2004. Borrowing their formal model for representing temporal expressions contained in documents (e.g., `in the summer of 2004`) and capturing their inherent uncertainty, we put forward a generative model for identifying time intervals of interest to a given keyword query. Our model is based on the intuition that a time interval of interest should be often referred to in relevant documents. More specifically, it considers the top-k documents retrieved by a unigram language model, treating them as pseudo-relevant, and analyzes their contents, specifically the temporal expressions therein, for often referred to time intervals. We describe the design space and consider different concrete instantiations of our model. To evaluate their performance, we compile two novel testbeds, consisting of temporally unambiguous and temporally ambiguous queries obtained from high-quality web sources.

Contributions made in this work are: (i) a novel approach to identify time intervals of interest to a given keyword query, (ii) two testbeds consisting of temporally (un) ambiguous queries which are made publicly available, (iii) an experimental evaluation of our approach on The New York Times Corpus [1], as a publicly-available document collection, on the aforementioned query testbeds.

Organization. The rest of this paper is organized as follows. We put our work in context with prior research in Section 2. Section 3 then describes our approach, including a discussion of the design space and details on our concrete instantiation. Following that, we describe our experimental evaluation in Section 4, before concluding in Section 5.

CIKM'14, November 3–7, 2014, Shanghai, China.

http://dx.doi.org/10.1145/2661829.2661927.

2. RELATED WORK

In this section, we put our work in context with existing prior work. Kanhabua et al. [9] is the work closest to ours. In contrast to the approach put forward in this work, their method focuses on identifying years of interest to a keyword query and does so only based on documents' publication dates. Their method is thus restricted to time points at year granularity and cannot identify time intervals at other granularities. Dakka et al. [4] as well as Diaz and Jones [7], as one building block in their respective research, describe methods that identify time points of interest to a query. Their methods, though, are solely based on the publication dates associated with documents and do not consider temporal expressions from their contents. Again, no time intervals are considered and the granularity is limited to that of documents' publication dates. Strötgen et al. [13] look into the related problem of identifying salient temporal expressions from a document. Other work has looked into improving the result quality of implicitly or explicitly temporal queries. For the former, this includes Metzler et al. [11], who identify implicitly temporal queries within the query log of a web search engine, and Dakka et al. [4], who analyze the distribution of publication dates to identify implicitly temporal queries. Peetz et al. [12] is a recent related work that leverages bursts in the temporal distribution of publication dates to improve retrieval effectiveness. Berberich et al. [3], as the work mentioned in the introduction, targets explicitly temporal queries and leverages both documents' publication dates and temporal expressions. Our work is orthogonal and the time intervals that we identify can be used to augment the query and obtain better results with one of the aforementioned approaches. Finally, there has been work on attaching a time point or time interval to an entire document. Thus, de Jong et al. [5] determine the likely publication time of a document based on its language; Kanhabua et al. [8] make use of temporal expressions from documents' contents to the same end. Jatowt et al. [6], even in the absence of any temporal expressions, determine a so-called focus time for a document, which delimits the time period the document predominantly refers to. For all of these approaches, the focus is on identifying a single time point or time interval (as opposed to possibly more than one) for a given document (as opposed to a query in our case).

3. IDENTIFYING INTERESTING TIME INTERVALS

In this section, we describe our approach for identifying interesting time intervals for a given keyword query.

3.1 Document Model

We largely adopt the formal model and notation introduced by [3]. Our document collection is denoted $\mathcal{D}$. A document $d \in \mathcal{D}$ consists of a multiset of keywords d_{text} and a multiset of temporal expressions d_{time}. We let $tf(v, d)$ and $tf(T, d)$ denote the term frequency of the keyword v and the temporal expression T in document d, respectively. We use $|d_{text}|$ and $|d_{time}|$ to denote the multiset cardinalities of the textual and temporal part, respectively. In the remainder, when it is clear from the context, we simply write d to refer to either of them. Keywords are drawn from a vocabulary $\mathcal{V}$. A temporal expression is a four-tuple

$$T = \langle\, tb_l,\, tb_u,\, te_l,\, te_u \,\rangle$$

with components from a time domain $\mathcal{T}$ (usually $\mathbb{N}$). Such a temporal expression can refer to any time interval $[tb,\, te] \in \mathcal{T} \times \mathcal{T}$ with $tb_l \leq tb \leq tb_u$ and $te_l \leq te \leq te_u$, i.e., tb_l (te_l) and tb_u (te_u) mark the earliest and latest begin (end) of such times intervals. This representation treats time intervals as having a precise meaning and captures the uncertainty inherent to temporal expressions such as `in the 1990s`, which at year-granularity would be mapped to $\langle\, 1990,\, 1999,\, 1990,\, 1999 \,\rangle$, thus potentially referring to any time interval completely within the decade. Alternatively, a temporal expression T can be regarded as a set of time intervals, namely all of the time intervals that it can refer to. We will use this interchangeably and, for instance, use $|T|$ as the number of time intervals the temporal expression refers to.

3.2 Retrieval Model

As mentioned above, our approach determines time intervals of interest to a query based on pseudo-relevant documents. To determine those, we use a unigram language model with Dirichlet smoothing and thus estimate the query likelihood of a given keyword query q as

$$P\,(\,q \mid d\,) = \prod_{v \in q} \frac{tf(v,d) + \mu \cdot \frac{tf(v,D)}{|D|}}{|d| + \mu}\ . \tag{1}$$

Here, D is the document collection, treated as a single document, for the purpose of smoothing probability estimates.

3.3 Time Intervals of Interest

Having identified documents believed to be relevant to the keyword query q, our approach analyzes their contents to determine time intervals of interest. We next describe the high-level components of our approach, before discussing possible instantiations.

Intuitively, a time interval $[tb,\, te]$ is considered interesting for a keyword query q, if it is frequently referred to by highly relevant documents. We cast this intuition into the following generative model:

$$P(\,[tb,\, te] \mid q\,) \;=\; \sum_{d \,\in\, top(q,k)} P(\,[tb,\, te] \mid d\,)\, P(\,d \mid q\,) \tag{2}$$

According to this model, first a document d is selected from $top(q, k)$ as the set of k documents having highest likelihood of generating the keyword query q. Second, a time interval $[tb,\, te]$ is generated from the temporal expressions contained in document d. For each of the two steps, we consider different design alternatives.

Generating Documents

In the simplest case, in the first step, a document is selected at uniform random among the top-k results, yielding

$$P(\,d \mid q\,) = 1/k\ . \tag{3}$$

Here, the query likelihood $P(\,q \mid d\,)$ is thus not taken into account. While this may not be a problem for small choices of k, we expect it to deteriorate performance for larger choices. As an alternative, we consider

$$P(\,d \mid q\,) = \frac{P(\,q \mid d\,)}{\sum_{d' \,\in\, top(q,k)} P(\,q \mid d'\,)}\ , \tag{4}$$

which estimates the probability of selecting a document in the first step as proportional to its query likelihood estimated according to Equation 1.

Generating Time Intervals

For the second step, we can estimate the probability of generating the time interval $[tb, te]$ from document d as

$$P(\,[tb,\,te]\mid d\,) = \frac{1}{|d_{time}|}\sum_{T\,\in\,d_{time}} \mathbb{1}(\,[tb,\,tb,\,te,\,te] = T\,)\;. \quad (5)$$

The time interval $[tb, te]$ can thus only be generated from documents containing temporal expressions that exactly map to it. To illustrate this, the time interval [1992, 1998] can only be generated from documents that contain `from 1992 until 1998` but not from documents containing only `in the 1990s`. As a more relaxed advanced alternative, building on the generative model introduced in [3], we also consider

$$P(\,[tb,\,te]\mid d\,) = \frac{1}{|d_{time}|}\sum_{T\,\in\,d_{time}} \frac{\mathbb{1}(\,[tb,\,te]\in T\,)}{|T|}\;, \quad (6)$$

which takes into account the uncertainty inherent to temporal expressions. With this model, also a document containing `in the 1990s`, formally represented as $\langle$ 1990, 1999, 1990, 1999 $\rangle$, could generate the time interval [1992, 1998].

Query Processing

At query time, our method first determines the set $top(q, k)$ of documents having highest query likelihoods. It then analyzes the temporal expressions therein, determining t_{min} and t_{max} corresponding, respectively, to the earliest and latest time mentioned in any of the result documents. Following that, it enumerates all valid time intervals $[tb, te] \subseteq [t_{min}, t_{max}]$ and determines their probability $P(\,[tb,\,te]\mid d\,)$. For this last step, combining the two design alternatives for each of the two steps of our generative model, we obtain four possible instantiations, which we experimentally evaluate in the following section. We will use **N** and **A** to refer to the *naïve* and *advanced* design alternative for each of the two steps. The method combining Equation 4 and Equation 5, for example, will be referred to as **AN**.

4. EXPERIMENTAL EVALUATION

We now describe our experimental evaluation of the approach put forward in this work.

4.1 Setup & Datasets

Document Collection. As a document collection, we use The New York Times Annotated Corpus [1], which consists of about 2 million news articles published between 1987 and 2007. Publication dates are readily available. Temporal expressions are obtained from the data provided by [3] – they used TARSQI [14] to annotate temporal expressions augmented by a handful of handcrafted regular expressions to go after range expressions (e.g., `from 1980 until 1984`). Publication dates of documents are taken into account as additional temporal expressions – thus a document published on March 13, 1988 virtually contains the temporal expression `on March 13, 1988`.

Queries. We use two sets of test cases: **(i)** *temporally unambiguous queries* obtained from the "On this Day" website of The New York Times[1]. For each day of the year, this website lists an event of historic significance, including a concise description. For example, for July 1st, the event is described as *"In 1997, Hong Kong reverted to Chinese rule after 156 years as a British colony."*. We extract the indicated year (here: 1997) for each date to obtain a precise date at day granularity and keep the rest of the description as a query. This leaves us with a total of 366 temporally unambiguous queries; **(ii)** *temporally ambiguous queries* from the domains of Sports, Music, Movies, Politics, and History, which we compiled manually. For each of them, we consult Wikipedia to find out the associated time intervals at day granularity. The obtained set of 20 queries is given in Table 3. Here, the number of associated time intervals is given in parentheses, indicating the degree of ambiguity of each query. In the interest of repeatability, both query sets, including associated time intervals are made available at:

`http://www.mpi-inf.mpg.de/~kberberi/data/cikm2014`

[1] `http://learning.blogs.nytimes.com/on-this-day/`

Sports	commonwealth games (21) | asian games (18) | summer olympics (34) | winter olympics (26) | super bowl winners (48)
Music	u2 album (13) | nirvana album (4) | beatles album (52) | red hot chilli peppers album (11) | michael jackson album (11)
Movies	harry potter movie (6) | oscar academy awards (88) | lord of the rings movie (3)
Politics	german federal elections (19) | us presidential elections (58) | australia federal elections (45)
History	iraq war (2) | world trade center bombing (2) | madrid bombing (9) | earthquake united states of america (73)

Table 3: Temporally ambiguous queries

Methods under comparison are the four combinations of the naïve and advanced models delineated in Section 3, referred to as **NN**, **AN**, **NA**, and **AA**. We can not sensibly compare against [9] as a baseline, since their method is based on publication dates and year granularity. For each of the methods under comparison, we set the smoothing parameter of the unigram language model as $\mu = 1000$ and vary the number of pseudo-relevant documents retrieved as $k = \{\,25,\,50,\,100\,\}$. We consider three different temporal granularities (day, month, year) in our experiments. When going for a coarser granularity (e.g., year), temporal expressions, which are natively stored at day granularity, are systematically coarsened. As a concrete example, the temporal expression $\langle$ 19980101, 19981231, 19980101, 19981231 $\rangle$ would be converted into $\langle$ 1998, 1998, 1998, 1998 $\rangle$ at year granularity. The same procedure is applied to the ground-truth time intervals of our query test cases.

Measures. We use Precision@k (P@k) as a measure of retrieval effectiveness. For the sake of comparability, we report P@1 and P@5 for both the unambiguous and ambiguous queries – instead of using mean reciprocal rank (MRR) for the unambiguous case.

4.2 Experimental Results

Table 1 shows values of P@1 and P@5 obtained for unambiguous queries. We observe relatively higher precision values for **NA** and **AA**, which rely on the advanced approach to estimate $P(\,[tb,\,te]\mid d\,)$. Both achieve similar performance, indicating that our advanced method to estimate $P(\,d\mid q\,)$, taking into account query likelihoods, is not effective. This is substantiated by the performance of **NN** and **AN** – while the latter uses the advanced method to estimate

	Day						Month						Year					
	P@1			P@5			P@1			P@5			P@1			P@5		
k	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**
NN	0.03	0.04	0.04	0.02	0.03	0.03	0.06	0.06	0.03	0.06	0.08	0.01	0.03	0.04	0.04	0.02	0.03	0.03
AN	0.03	0.03	0.04	0.02	0.03	0.03	0.06	0.05	0.03	0.06	0.08	0.01	0.02	0.01	0.01	0.01	0.01	0.01
NA	0.07	0.06	0.09	0.04	0.04	0.04	0.18	0.18	0.18	0.10	0.10	0.05	0.14	0.17	0.10	0.11	0.11	0.08
AA	0.06	0.06	0.09	0.04	0.04	0.04	0.19	0.17	0.20	0.09	0.10	0.07	0.14	0.17	0.10	0.11	0.11	0.08

Table 1: Temporally unambiguous queries

	Day						Month						Year					
	P@1			P@5			P@1			P@5			P@1			P@5		
k	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**	**25**	**50**	**100**
NN	0.05	0.00	0.00	0.01	0.01	0.01	0.10	0.10	0.16	0.05	0.04	0.05	0.55	0.55	0.26	0.31	0.36	0.33
AN	0.05	0.05	0.05	0.01	0.01	0.02	0.10	0.15	0.16	0.05	0.05	0.05	0.60	0.60	0.32	0.35	0.34	0.34
NA	0.10	0.10	0.16	0.05	0.10	0.10	0.35	0.50	0.42	0.25	0.26	0.31	0.75	0.75	0.74	0.59	0.58	0.54
AA	0.10	0.10	0.16	0.04	0.10	0.10	0.35	0.50	0.42	0.25	0.26	0.31	0.75	0.75	0.74	0.59	0.58	0.54

Table 2: Temporally ambiguous queries

$P(d \mid q)$, its precision values are as low as those obtained by the completely naïve **NN**. It can also be seen that methods' performance varies with temporal granularity, peeking at month granularity. Finally, we observe that considering more pseudo-relevant documents only pays off to a point – for none of the methods performance increases consistently as we go beyond $k = 50$.

Results for ambiguous queries are shown in Table 2. All four methods consistently achieve higher values of P@1 and P@5 than for the unambiguous case. Comparing **NN** and **AN**, we again observe that the advanced method of estimating $P(d \mid q)$ is not very effective. In contrast, we see good improvements for **NA** and **AA**, indicating that the more advanced handling of temporal expressions pays off. For ambiguous queries, as a difference from the unambiguous case, we observe that all methods achieve their best performance for year granularity. However, again we do not see consistent improvements as more pseudo-relevant documents are considered for larger choices of k.

Summary

Our experiments, using temporally unambiguous and temporally ambiguous queries as test cases, have shown that **NA** and **AA** perform similarly and are ahead of the other two configurations. Thus, the advanced method to handle temporal expressions and estimate $P([tb, te] \mid d)$ is effective; the advanced method to estimate $P(d \mid q)$, on the other hand, has no effect.

5. CONCLUSION

We have proposed a novel approach to identify time intervals of interest for a given keyword query. Our approach is based on a generative model and we considered four possible instantiations of it. Experiments on temporally unambiguous queries and temporally ambiguous queries as test cases showed that there are effective instantiations of our approach – considering temporal expressions and their inherent uncertainty pays off; factoring in query likelihoods does not. As part of our future research, we plan to investigate (i) how users perceive the interestingness of the determined time intervals and (ii) how retrieval effectiveness is affected when using the determined time intervals in query expansion.

6. REFERENCES

[1] The New York Times Annotated Corpus `http://corpus.nytimes.com`.

[2] O. Alonso, M. Gertz, and R. A. Baeza-Yates. On the value of temporal information in information retrieval. *SIGIR Forum*, 41(2):35–41, 2007.

[3] K. Berberich, S. Bedathur, O. Alonso, and G. Weikum. A language modeling approach for temporal information needs. In *ECIR*, 2010

[4] W. Dakka, L. Gravano, and P. G. Ipeirotis. Answering general time-sensitive queries. *IEEE Trans. Knowl. Data Eng.*, 24(2):220–235, 2012.

[5] F. M. G. de Jong, H. Rode, and D. Hiemstra. Temporal language models for the disclosure of historical text. In *AHC*, 2005

[6] A. Jatowt, C. Man Au Yeung, and K. Tanaka. Estimating document focus time. In *CIKM*, 2013.

[7] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Trans. Inf. Syst.*, 25, 2007.

[8] N. Kanhabua and K. Nørvåg. Using temporal language models for document dating. In *ECML/PKDD*, 2009.

[9] N. Kanhabua and K. Nørvåg. Determining time of queries for re-ranking search results. In *ECDL*, 2010.

[10] X. Li and W. B. Croft. Time-based language models. In *CIKM*, 2003.

[11] D. Metzler, R. Jones, F. Peng, and R. Zhang. Improving search relevance for implicitly temporal queries. In *SIGIR*, 2009.

[12] M.-H. Peetz, E. Meij, and M. de Rijke. Using temporal bursts for query modeling. *Inf. Retr.*, 17(1):74–108, 2014.

[13] J. Strötgen, O. Alonso, and M. Gertz. Identification of top relevant temporal expressions in documents In *TempWeb* 2012.

[14] M. Verhagen, I. Mani, R. Sauri, J. Littman, R. Knippen, S. B. Jang, A. Rumshisky, J. Phillips, and J. Pustejovsky. Automating temporal annotation with tarsqi. In *ACL*, 2005.