TECHNICAL REPORT

YL-2009-002

AN ANALYSIS OF IMPLICITLY YEAR QUALIFIED QUERIES

Donald Metzler, Rosie Jones, Fuchun Peng, and Ruiqiang Zhang
Yahoo! Labs
{metzler, jonesr, fuchun, ruiqiang}@yahoo-inc.com

Bangalore • Barcelona • Haifa • Montreal • New York
Santiago • Silicon Valley

Yahoo! Labs Technical Report No. YL-2009-002

Yahoo! Labs Technical Report No. YL-2009-002

AN ANALYSIS OF IMPLICITLY YEAR QUALIFIED QUERIES

Donald Metzler, Rosie Jones, Fuchun Peng, and Ruiqiang Zhang
Yahoo! Labs
{metzler, jonesr, fuchun, ruiqiang}@yahoo-inc.com

ABSTRACT: A significant fraction of search queries relate to topics that are time dependent, where the ideal result changes from one time-span to the next. Despite their prevalance and importance, such queries have been largely ignored. In this paper[1], we focus on a subset of temporal queries known as implicitly year qualified queries. Such queries, which include "olympics" and "miss universe" implicitly depend on the year. We describe an efficient algorithm for automatically mining these queries using query log analysis. We apply our algorithm to a large query log consisting of over 750 million web search queries from a commercial search engine and analyze the characteristics of the mined queries. Finally, we show that web search effectiveness for certain classes of implicitly year qualified queries can be significantly improved using our proposed temporal result set reordering approach.

[1]This technical report is an extended version of [15].

1. Introduction

Many web search queries have implicit intents associated with them that, if detected and used effectively, can be used to improve search quality. For example, a user who enters the query "toyota camry" may wish to find the official web page for the car, reviews about the car, or the location of the closest Toyota dealership. However, since the user only entered a couple of keywords, it may be difficult to accurately determine which of these implicit intents the user actually meant. Given such an ambiguous query, a search engine must use personalization [7, 23], click information [18, 22], query log analysis [3, 5, 17], and other means for determining implicit intent.

Rather than solving the general problem of automatically determining user intent, we focus on queries that have a temporally dependent intent. Temporally dependent queries are queries for which the best search results change with time. Simple examples include "new years" and "presidential elections", which are events that recur over time. The search results for these queries should reflect the freshest, most current results. A slightly more complex example is the query "turkey". For this query, it may be useful to return turkey recipes or cooking instructions around the Thanksgiving holiday and travel information during peak vacation times. In all of our examples thus far, the events have occurred with (mostly) predictable periodicity. However, for queries such as "oldest person alive", the best result changes unpredictably, making it difficult for search engines to consistently return correct results. Therefore, temporally dependent queries come in many different forms and pose many challenges to search engines.

In this paper, we investigate a subset of temporal queries that we call *implicitly year qualified* queries. A year qualified query is a query that contains a year. An implicitly year qualified query is a query that does not actually contain a year, but yet the user may have implicitly formulated the query with a specific year in mind. An example implicitly year qualified query is "miss universe". It is plausible that the user actually meant "miss universe 2008", "miss universe 2007", or maybe even "miss universe 1990", yet did not actually qualify the query with a year. Other examples include "olympics", "toyota camry", "easter", and the names of conferences, such as "SIGMOD" or "KDD".

Implicitly year qualified queries are particularly interesting and challenging from a search point of view. Our later analysis will show that more than *7%* of queries belong to this category. However, correctly handling this type of query is not trivial. As anecdotal evidence, as of this writing, only one of the three major search engines ranks the SIGIR 2009 web page higher than the SIGIR 2008 web page for the query "SIGIR", despite the fact that most people searching for the SIGIR conference are interested in the upcoming event, rather than the previous event. Therefore, in order to improve search quality for temporal queries, a search engine must be able to detect that certain queries have an implicit temporal intent and use this information to improve the search results. These are precisely the types of issues that we tackle in the remainder of this paper.

The primary contributions of this paper concern three aspects of implicitly year qualified queries. First, our work, which is mostly exploratory, investigates many unique aspects of temporal queries that have not been deeply analyzed in the past. Second, we propose a simple, efficient algorithm for automatically mining such queries using query log analysis. We apply our algorithm to a query log consisting of over 750 million queries from a commercial search engine and provide a detailed analysis of the results. Finally, we propose using temporal result set reordering to improve web search result quality for implicitly year qualified queries. Our experimental results show the proposed approach significantly improves retrieval effectiveness for ACM conference name queries, as well as general open-domain web queries.

The remainder of this paper is laid out as follows. First, in Section 2 we discuss related work. Then, in Sections 3 and 4 we detail our algorithms for mining implicitly year qualified queries and analyze the properties of queries mined from the query logs of a commercial search engine. Next, Section 5 explains how the temporal information mined using our algorithms can be used to improve search quality. Finally, in Section 6 we conclude the paper and discusses possible directions for future work.

2. Related Work

We consider related work of three kinds: those relating to temporal patterns of documents sets, those relating to temporal patterns of query frequency, and those relating to query enrichment using query log analysis.

2.1. Temporal Information in Documents

Some recent research has looked at documents retrieved in response to a query, and the timelines which can be generated from the timestamps on those documents. Li and Croft [12] look at re-ranking documents with a time-based document prior, preferring documents with recent timestamps. They conducted their experiments with queries whose relevant documents were all recent in the collection, showing that it is useful to prioritize recent documents in such a case. Diaz and Jones [6] and Dakka et al. [4] considered timelines produced by the timestamps of documents returned in response to a query, to predict the temporal ambiguity or focus of a query. Mei et al. [14] showed how we can model the shifts in interest in a topic over time and space, by modeling the location and timestamps of blog stories retrieved in response to a query on that topic. Kleinberg [11] modeled how word distributions vary over time in a stream of documents, and that bursts of word frequency correspond to new emerging topics. Finally, Amitay et al. [1] show that trends can be detected by approximating document timestamps using link analysis techniques.

Our work differs from these in that we detect implicit temporal intent by examining queries, rather than documents. This allows us to avoid problems that may arise from inaccurate or missing document timestamps.

2.2. Temporal Information in Query Frequency Distributions

Other work has looked at timelines built from the frequency of search queries over time, which reflects how user interest in a topic varies over time.

Chien and Immorlica [2] showed that queries which have similarly frequency patterns have similar meaning. Liu et al. [13] show that queries which are similar in time series are not necessarily those that co-occur in user search sessions. Jiang et al. [9] showed that increases in query trends over time almost always correspond to real world events.

Our proposed algorithm differs from these previous approaches in that we look at temporal expressions in queries, such as years, rather than the temporal distribution of queries. Previous approaches are limited by the amount of query log data available. For example, if query log data is only available from 2005 to 2008, then these previous approaches would only be able to detect that the "olympics" was an event of interest in 2006 and 2008, but it would fail to detect that many users are actually interested in the 2010 Vancouver or 2012 London olympics. It would also fail to detect any interest in the olympics prior to the beginning of their query log data (2005), which limits the usefulness of the approaches. However, by considering the actual text of queries, we are able to have a much more comprehensive understanding of user's implicit temporal intents that goes well beyond the timespan of the query log.

2.3. Query Enrichment using Query Logs

Cui et al. [3] and Peng et al. [17] showed that web search quality can be improved by automatically rewriting or expanding queries to contain related terms using query log analysis. Both of these approaches enrich the original query representation by augmenting it with information mined from the query log.

In this work, we adopt a similar approach by enriching implicitly year qualified queries with implicit temporal intent information mined from a query log. Unlike the previous approaches, however, our approach is specific to temporal queries, and rather than solving the vocabulary mismatch problem, we attempt to solve the more specific *temporal mismatch* problem. Since the problem is more specific, we are able to develop more specialized solutions.

3. Mining Implicitly Year Qualified Queries

We will now describe our proposed algorithm for automatically mining implicitly year qualified queries using query log analysis. Our algorithm is only designed to detect implicitly year qualified queries. It does not automatically classify the mined queries into any type of taxonomy. However, a variety of query classification approaches [21] can be used to map the mined queries into a taxonomy, if necessary.

3.1. Algorithm

Our proposed algorithm is simple, yet efficient. It relies only on having access to a query log with frequency information. The approach does not rely on any user, click, or session information. The approach was designed with simplicity in mind so that it could be efficiently applied to very large query logs for exploratory purposes.

The primary idea behind our proposed approach is to see how often a base query, such as "olympics" is year qualified in the query log. If the base query is commonly year qualified, then we declare that it is implicitly year qualified. For the olympics case, it is very likely that the queries "olympics 2008", "2004 olympics", etc. occur with relatively high frequency in the log, which allows us to determine that "olympics" is implicitly year qualified. Thus, the foundations of our algorithm are built upon the following assumptions:

- Implicitly year qualified queries are strongly associated with several different years.
- Implicitly year qualified queries are associated with years more than they are associated with non-years.

We now describe how these two assumptions can be turned into an effective mining algorithm.

Perhaps one of the most important properties of implicitly year qualified queries is how strongly they are associated with a given year. This value, which we call the year qualified weight, is formally defined as:

$$w(q, y) = \#(q.y) + \#(y.q) \tag{3.1}$$

where $\#(q.y)$ denotes the number of times that the base query q is post-qualified with the year y in the query log. Similarly, $\#(y.q)$ is the number of times that q is pre-qualified with the year y. Examples of year pre- and post-qualifications are "world series 2008" and "2008 world series", respectively.

The year qualified weight measures how likely q is to be qualified with y, which forms the basis of our mining and analysis. We detect years in queries using the regular expressions `19[0-9][0-9]` and `20[0-9][0-9]`. Although simple, we find that these regular expressions actually produce very high precision and high recall year detections. Most of the false positives are product numbers, while most of the false negatives correspond to one-time historical events, such as "war of 1812", which are of less interest to us than more current events.

Given these weights, we use the following methodology to automatically mine implicitly year qualified queries. Given a query q, we first compute $w(q, y)$ for all plausible years. We then use the following function to determine if a query is implicitly year qualified:

$$isIYQQ(q) = \begin{cases} 1 & |\{y : w(q, y) > 0\}| \geq 2 \\ 0 & otherwise \end{cases}$$

which simply states that a query is implicitly year qualified if it is qualified by at least two unique years. We set the threshold at 2 because we are interested in temporally recurring events. However, it is possible to mine one-time events by lowering the threshold to 1, although this may also introduce a significant number of spurious or noisy detections.

Even though a query is identified as implicitly year qualified does not necessarily mean that the query should always be treated as temporal in nature. Consider the query "chi", which happens to be the name of a popular human-computer interaction conference. This query is clearly temporal, and indeed, our algorithm detects the query as being implicitly year qualified. However, "chi" is a very common term that is often qualified in many different ways, including "chi squared" (statistical test), "chi chis" (restaurant), and "chi omega" (sorority). In fact, these other qualifications are much more common than the temporal ones. We call this phenomenon *temporal ambiguity*. Implicitly year qualified queries, such as “chi” that are associated with many different qualifications are temporally ambiguous, whereas other queries, such as “sigir” that are almost exclusively reformulated or qualified with a year are deemed temporally unambiguous. We quantify temporal ambiguity as follows:

$$\alpha(q) = \frac{\sum_y w(q,y)}{\sum_x \#(q.x) + \sum_x \#(x.q)} \tag{3.2}$$

where the sums $\sum_x \#(q.x)$ and $\sum_x \#(x.q)$ go over all pre- and post-qualifications for the query q. It should be easy to see that if the query is always qualified with a year then $\alpha(q) = 1$. Although we call this measure temporal ambiguity, it may also be interpreted as a confidence value that the query has implicit temporal intent.

3.2. Discussion

It may be possible to extend our approach to use query reformulation logs instead of query logs for mining [10]. In this scenario, if a user specifies a query, then rewrites the query and adds (or deletes) a year, it would give us valuable information for determining that the original query is implicitly year qualified. Indeed, this would provide a causal link that cannot be inferred from query logs alone. Such an approach requires session information, session boundary detection, and is more computationally expensive than our approach. Although not explored here, this is an interesting potential direction for future work.

It is also important to reiterate that the goal of this paper is to undertake a preliminary investigation into detecting temporal intent. We focus on implicitly year qualified queries simply as a first step to examine the feasibility of the simplest class of implicitly temporal queries. Although beyond the scope of the current work, we believe that our proposed approach, with some modifications, can be used to detect more general notions of implicit temporal intent.

4. Analysis of Mined Queries

In this section, we perform a detailed empirical analysis of the queries mined using our proposed algorithm. For our analysis, we use 18 months of query log data that consists of about 750 million web search queries. The queries in the data set were normalized, scrubbed of personal information, and filtered to eliminate singleton, adult, and “junk” queries.

4.1. Types of Implicitly Year Qualified Queries

We begin by describing the types of queries that our algorithm discovered. The queries can be described with respect to two dimensions. The first dimension is *topicality*, which characterizes what the query is about. We found that a majority of implicitly year qualified queries fall into one of the following topical categories: auto (“chevrolet malibu”), awards (“grammy awards”), convention or conference (“dental seminars”), date or calendar (“triathalon calendar”), entertainment (“american idol”), exam (“sats results”), government (“tax forms”), holiday or occasion (“national teacher’s day”), information or research (“building codes”), list or ranking (“top 25 hip hop songs”), local event ("east durham irish festival"), news ("hurricane"), pop culture ("clay aiken girlfriend"), product ("burburry sunglasses"), report or guide ("us news college guide"), seasonal program ("crump law camp"), sports ("british open"), and statistics ("life expectancy").

The second dimension attempts to characterize the temporal nature of the query with respect to its *periodicity*. A query is periodic if it recurs with a fixed frequency, such as every day, every year, every other year, every decade, etc. All other temporal queries, which are not related to periodic events are sporadic. These queries typically correspond to events that only occur a small number of times, such as concerts, or recurring events that happen unpredictably, such as hurricanes or flu pandemics.

We manually analyzed a sample of 400 queries mined using our technique and classified them according to their topicality and periodicity.

Our analysis reveals that periodic queries significantly outnumber sporadic queries. Approximately 70% of the queries analyzed were periodic. This results from the fact that periodic events are more likely to be year qualified than sporadic events. For example, a user is much more likely to search for "presidential elections 2000" than "hurricane katrina 2005".

Another observation is that entertainment, local events, and sports are the most common temporal topics, making up 13.5%, 13.5%, and 12% of the analyzed queries, respectively. Other popular temporal topics include lists and rankings (8.5%), auto (6%), statistics (4.5%), and products (5.5%), which mostly come in the form of software versions ("microsoft office 2007").

Our analysis also indicates that certain topics are always periodic or always sporadic by their very nature. For example, new cars tend to be produced annually. Awards, exams, local events, reports, seasonal programs, and sporting events also tend to recur cyclically. On the other hand, news events and pop culture references are more sporadic and do not occur according to a fixed schedule. The remaining topical categories can either be periodic or sporadic. For example, in the entertainment category, television shows tend to recur annually, while concerts and movies occur sporadically.

Therefore, as our analysis shows, implicitly year qualified queries are very diverse. They span a number of topical categories and have varied temporal characteristics. This poses significant challenges when trying to utilizie the mined queries for improving search, as we will show later in this paper.

4.2. Frequency Distribution

Second, we investigate the query frequency distribution of the mined queries. Figure 1 is a log-log plot of the number of queries mined versus query frequency. As the figure shows, the distribution clearly follows a power law, indicating that most of the queries mined have very low query frequency while a small number of queries mined have a very large query frequency. This observation is consistent with previous work that has shown query frequency of web queries, in general, follows a similar distribution. Overall, we found that implicitly year qualified queries make up approximately 7% of the query volume.

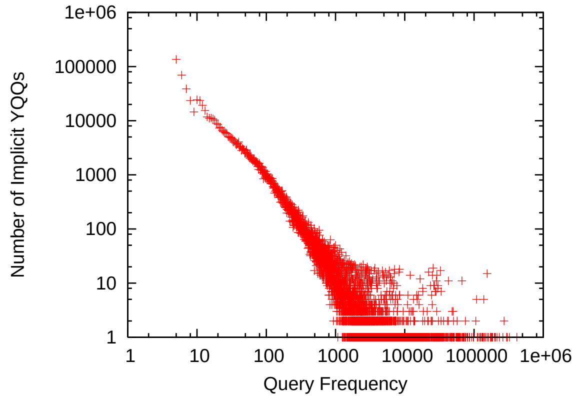

Figure 1: Log-log plot indicating the power law that exists between query log frequency and the number of year qualified queries with that frequency.

4.3. General Characteristics

Third, we look at the general characteristics of the mined queries. In order to do so, we examine the following, with the results of this analysis are shown in Table 1:

1. The mined queries that are associated with the most unique years (i.e., have the largest $|y : w(q, y) > 0|$),
2. The queries that are most frequently year qualified (i.e., have the largest $\sum_y w(q, y)$), and
3. The queries that are the most densely year qualified (i.e., have the largest $\frac{\sum_y w(q,y)}{|y:w(q,y)>0|}$).

As the results show, "january 1" is the query associated with the most years (109) in our data set. Not surprisingly, most of the queries associated with many years are specific dates, which tend to be holidays or historically relevant. The two exceptions are the queries "calendar year" and "world events" which are generic queries that are often qualified with a year.

As for the most frequently qualified queries, we see a completely different list, since dates are not queried for very often. In this column, we find queries associated with recurring international events, such as "miss universe" and "world cup", as well as holidays and popular television shows.

Finally, the most dense column shares some overlap with the most frequent column. These queries are those that are popularly queried for over the entire span of their existence. As the list indicates, users are confused, year after year, about when to change their clocks for daylight saving time. Users also want to know when easter is and enjoy finding the latest information about their favorite television shows.

Most Years	Most Frequent	Most Dense
january 1 (109)	miss universe (898521)	daylight savings time (62318)
april 14 (108)	easter (734949)	easter (66813)
december 12 (108)	world cup (693490)	dancing with the stars (62318)
december 21 (106)	calendar (635724)	american idol (49278)
calendar year (105)	daylight savings time (629800)	windows (44452)
august 1 (104)	wwe smackdown vs raw (483939)	wwe smackdown vs raw (43994)
june 14 (102)	madden (393877)	black friday (42846)
april 23 (100)	super bowl (390426)	calendar (42381)
world events (100)	microsoft office (382690)	madden (39387)
april 1 (99)	dancing with the stars (373909)	daylight saving time (35096)

Table 1: The empirically mined year qualified queries that are associated with the most years, most frequently year qualified, and have the highest year qualification density.

4.4. Implicit Temporal Profiles

Next, we analyze the temporal data mined from general web search queries. We begin by examining the year qualified weights (Equation 3.1) of the mined queries. Figure 2 shows the distribution over year qualified weights for several different web search queries.

The first query is “january 1”, which, as we showed earlier, is the query temporally tied to the most years according to our algorithm. The algorithm shows that more recent years are the most popular, with several spikes in the past. Interestingly, one of the spikes corresponds to “january 1 1970”, which is the beginning of the Unix time epoch. The query “miss universe”, has a large amount of interest in this year’s and last year’s events, but increasing disinterest in previous events. Furthermore, the plot for “olympics” shows a clear every other year trend.

The other three queries are more interesting, because they do not follow such regular, periodic trends. The distribution for the first query, “ford mustang” (a car model) has two primary modes – one in the late 1960’s s and another in the early 2000’s. The first generation of Mustang’s, which are now considered classic cars, were first sold in the late 1960’s. The 1970’s, 1980’s, and 1990’s models were nowhere near as popular with car enthusiasts. However, the introduction of the 2005 model of the car garnered great attention, which accounts for the large spike around that year. Similarly, the “elton john tour” distribution accurately reflects the years that the artist went on tour, first in the late 1970’s, again in the early 1990’s, and yet again more recently. The last example, “flu pandemic”, also shows how this analysis can be meaningfully mapped to world knowledge and events. The peaks in the distribution, with the exception of the one around 2003, which is likely the result of avian flu fears, occur around 1918, 1957, and 1968, which correspond to the well-known Spanish, Asian, and Hong Kong flu pandemics, respectively.

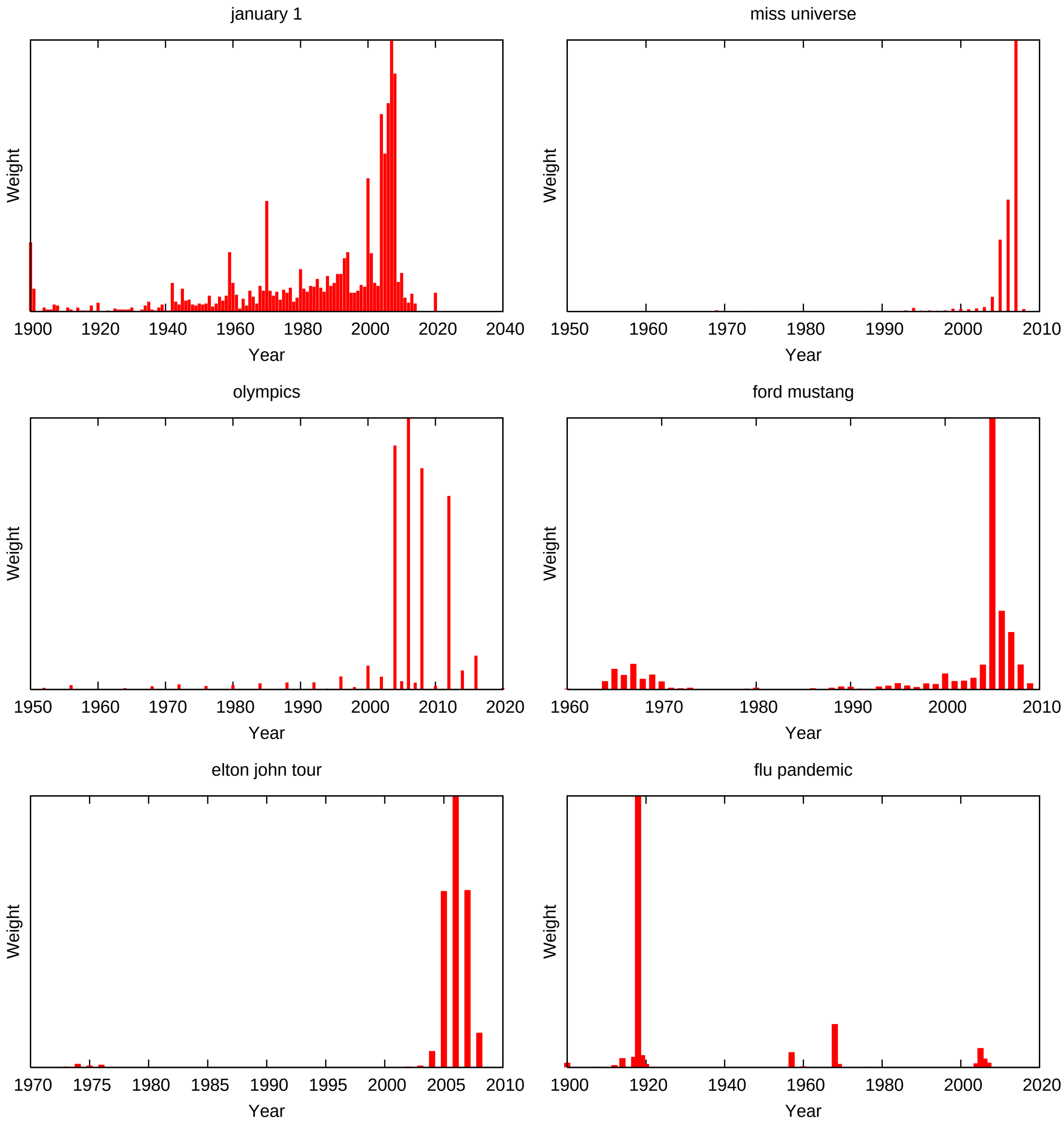

Figure 2: Implicit year distributions mined for the queries "january 1", "miss universe", "olympics", "ford mustang", "elton john tour", and "flu pandemic" (left to right, top to bottom).

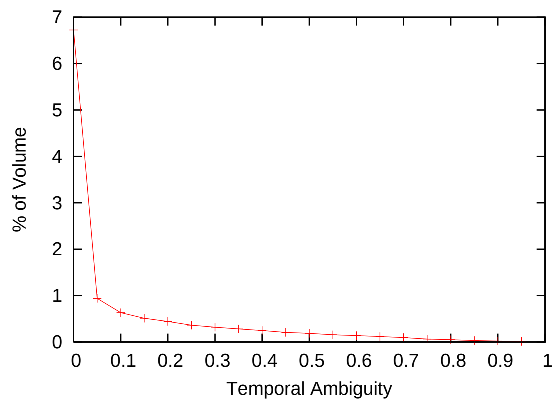

Figure 3: Query volume percentage as a function of temporal ambiguity.

4.5. Temporal Ambiguity

Finally, we explore the notion of temporal ambiguity in more detail. Figure 3 plots query volume as a function of temporal ambiguity. For each temporal ambiguity level α, we plot the volume of queries with temporal ambiguity at least α. The plot drops off very quickly, suggesting that many of common implicitly year qualified queries are temporally ambiguous (i.e., $\alpha(q)$ is low). Intuitively this make sense, since the most popular queries are likely to be rewritten in many different ways, in conjunction with many different terms, many of which will not be years, which results in low values of $\alpha(q)$. We will return to the issue of temporal ambiguity in the next section when we discuss its effect on temporal reordering of search results.

5. Application: Search

Now that we have a better understanding of the space of implicitly year qualified queries, showed that they can easily be mined, and analyzed their properties, we are ready to describe how the mined information can be used to improve search result freshness and quality for these types of queries. Our proposed technique attempts to bias search result sets towards documents that contain highly weighted years that are implicitly associated with the query, while also taking temporal ambiguity and recency into account, as well.

5.1. Temporal Result Set Reordering

Our proposed approach explicitly adjusts the score of document d in response to query q according to the years contained in the document. Given an implicitly year qualified query q, we first weight the qualified years associated with q as follows:

$$z(q, y) = \mathcal{N}(y; \mu, \sigma^2) \cdot \alpha(q) \cdot \frac{w(q, y)}{\max_y w(q, y)}$$

where $\mathcal{N}(y; \mu, \sigma^2)$ is a normal distribution with mean μ and variance σ^2, $\alpha(q)$ is the temporal ambiguity (Equation 3.2), and $w(q, y)$ is the number of times that query q is qualified with year y (Equation 3.1).

The weighting $z(q, y)$ has three primary components, each of which models a different important aspect of temporal queries. The first component, $\mathcal{N}(y; \mu, \sigma^2)$, models the *a priori* preference for a given year. In this paper we use a normal distribution with $\mu = 2008$ and $\sigma^2 = 1$ in order to impose a preference for recent years. However, the normal distribution can be replaced with any reasonable distribution or weighting to impose other reasonable types of preferences. Indeed, this value can be set to a constant, which would correspond to a uniform preference over years. The second component, $\alpha(q)$, which is the temporal ambiguity, models our confidence that the given query is temporal. If $\alpha(q)$ is large, then the query is very likely, based on query log analysis, to be temporal, and therefore it will receive a higher weight. The third and final component, $\frac{w(q,y)}{\max_y w(q,y)}$, is the normalized year qualified weight for the year as estimated from the query log. Thus, recent years that have large year qualified weights for temporally unambiguous queries will have large $z(q, y)$ weights.

Given a ranking function $S(q, d)$ that produces a score for documents d in response to query q, we can use the $z(q, y)$ weights to temporally bias $S(q, d)$ towards q's implicit temporal intent as follows:

$$S'(q, d) = S(q, d) + \sum_{f \in d} \lambda_f \sum_{y \in f} z(q, y)$$

where $S'(q, d)$ is the temporally biased score, $S(q, d)$ is the original score of document d with respect to query q, $f \in d$ is the set of document fields (e.g., title, anchor text, body), λ_f is the weight associated with field f, and $y \in f$ denotes the set of years that occur in field f in document d. Documents are then reordered according to $S'(q, d)$ to produce the final temporally-biased ranking.

This approach is similar to approaches commonly used in information retrieval to overcome the so-called vocabulary mismatch problem [20]. Here, rather than vocabulary mismatch, we have a temporal-mismatch problem, which is solved by reordering the result set based on temporal vocabulary terms (i.e., years). It is also important to note that the approach weights temporal matches according to the field the match occurs in, which has previously been shown to be more effective than ignoring the document structure [16, 19]. This allows us to boost the score of documents that match the year in the title or anchor text more than those that may spuriously match the year in the body. In our experiments, the title, (incoming) anchor text, body, and url weight are assigned weights (λ_f) of 2.0, 2.0, 0.5, and 0.5, respectively.

5.2. Results

We carried out two studies to determine the effectiveness of our proposed temporal reordering strategy. The first study looks at a set of closed domain queries (ACM conference names), while the second looks at a set of open domain queries (general web queries).

Ambiguity (α)	O > R	O < R	O = R
0.00 - 0.25	2	7	81
0.25 - 0.50	0	4	7
0.50 - 0.75	5	9	3
0.75 - 1.00	2	14	6
Total	9	34	97

Table 2: Temporal reordering results for conference queries. O is the original ranked list and R is the temporally reordered ranked list.

5.2.1. Closed Domain Queries Our first study considers conference name queries, which are a specialized subset of implicitly year qualified queries of particular interest to researchers. We gathered a list of 273 ACM conferences and workshops and used their acronyms, such as WSDM or SIGMOD, as queries that were then run through our mining algorithm. Our algorithm detected 142 of the conferences as implicitly year qualified queries, resulting in a recall of 52%. An error analysis revealed that most of conferences our algorithm failed to detect were caused by the filtering done on the queries and the fact that some of the conferences or workshops were 'retired' and had not been active, or searched for, over the past 18 months.

For each query, we presented a domain expert (i.e., researcher) with a list of the top 5 documents retrieved using a commercial search engine (circa July 2008) as well as the top 5 documents retrieved using our proposed temporal re-ranking strategy. Our re-ranking strategy is based on the $S(q, d)$ returned by the search engine. We then asked the expert to choose which ranked list contained the most relevant results with respect to the query. The expert also had the option of saying the two lists were equivalent in the case that the ranked lists were identical or neither list was clearly superior.

A summary of the results, stratified according to temporal ambiguity, is given in Table 2. As the results show, our reordering approach is preferred to the original ranking for 34 queries, was not preferred for 9 queries, and was deemed equivalent for the remaining 97 queries. The number of queries preferred using our temporal reordering approach over the baseline is statistically significant, thereby verifying the effectiveness of the method for this closed domain task.

The results, broken down by temporal ambiguity, show that the domain experts tended to prefer the temporally reordered results over the original results more for less temporally ambiguous queries. This is expected, as these are the queries that we expect to help the most using our proposed approach. Also, rather interestingly, nearly all of the queries with temporal ambiguity less than 0.25 (very ambiguous conference names) resulted in a "O=R" judgment from the domain experts. It is likely that these conference names are so ambiguous that the weights associated with the years are very low and have only a negligible impact on the final ranking, which is a desirable property. Examples of such queries include "chi", "colt", "paste", and "vast".

For the sake of illustration, we show the original and temporally reordered ranked lists for the query "CIKM" in Table 3. We see that the original ranked list contains results for CIKM 2002 and CIKM 2004, which are rather dated. The temporally reordered list promotes results for the 2008

Original Ranked List	Temporally Reordered Ranked List
Conference on Information and Knowledge Management (CIKM) www.cikm.org	CIKM 2008 — Home cikm2008.org
CIKM 2008 — Home cikm2008.org	Conference on Information and Knowledge Management (CIKM) www.cikm.org
Conference on Information and Knowledge Management (CIKM'02) www.cikm.org/2002	ACM CIKM 2007 - Lisbon, Portugal www.fc.ul.pt/cikm2007
CIKM www.informatik.uni-trier.de/ ley/db/conf/cikm/...	CIKM 2008 www.cikm.org/cikm-2008.html
CIKM 2004 ir.iit.edu/cikm2004	CIKM www.informatik.uni-trier.de/ ley/db/conf/cikm/...

Table 3: Top 5 results for the ACM conference query "CIKM". The original ranked list is on the left and the results of using temporal result set reordering are on the right.

Ambiguity (α)	Baseline	Temporal Reordering
0.00 - 0.25	10.09	10.06 (-0.31%)
0.25 - 0.50	8.48	8.46 (-0.15%)
0.50 - 0.75	7.67	7.69 (+0.32%)
0.75 - 1.00	7.02	7.07 (+0.84%)

Table 4: Baseline and temporal reordering results for open domain web queries.

conference and also includes a result for the 2007 conference, as well. The results for the 2002 and 2004 conferences are no longer in the list, considerably improving the perceived quality and freshness of the ranked list.

5.2.2. Open Domain Queries Our second study involves a set of 670 open domain web queries, which is considerably more heterogenous than the ACM conference name queries. As in our previous study, we compare our temporal reordering approach to the ranking of a commercial search engine. In this study, rather than asking domain experts to choose which ranked list is better, we asked a set of non-specialist editors to judge the relevance of each document retrieved for each query as either Perfect, Excellent, Good, Fair, or Bad. Since we are dealing with graded relevance judgments, we evaluate the baseline and our approach using DCG@5 [8].

Table 4 shows the results of the study. The first thing to notice is that DCG@5 decreases monotonically as the queries become less temporally ambiguous. This suggests that web search engines tend to perform substantially worse on temporal queries, highlighting the need for special temporal query handling.

Second, the results show that our proposed temporal result set reordering tends to improve temporally unambiguous queries more than temporally ambiguous queries. Similar results were found in the ACM conference query study, thereby reinforcing the applicability of our approach to temporally unambiguous queries.

Finally, we note that the 0.84% improvement observed for the 0.75-1.00 range is statistically significant, while none of the other differences are. These results suggest that our method can safely be applied to open domain web search queries that are identified as having a temporal ambiguity factor greater than 0.5. Although the improvements observed here are small (but statistically significant), we suspect that even larger improvements can be obtained by careful parameter tuning or more sophisticated query expansion techniques. Our goal was simply to show such implicit temporal information was useful, even when used in a simple manner, to reorder documents.

5.3. Discussion

In summary, our results show clear trends across two different studies, each with a different evaluation methodology. The primary outcomes of the studies are: 1) implicitly temporal queries are more difficult than non-temporal queries, primarily because search engines do not adequately take into account the implicit temporal aspect of the query, 2) our proposed temporal result reordering is most effective on closed domain queries, but can also be effective for temporally unambiguous open domain queries, and 3) it is generally easier to formulate a temporal query ranking strategy for closed domains than open domains, which is expected.

Therefore, we can conclude that there is likely no "one size fits all" ranking strategy for dealing with implicit temporal intent. Instead, each class or type of implicit temporal query likely should be handled differently. For example, a search engine should handle local event queries differently than news or product queries.

6. Conclusions and Future Work

In this paper we described the class of queries known as implicitly year qualified queries and proposed a simple yet efficient approaches for mining such queries using query log analysis. Furthermore, we conducted a detailed empirical analysis of the queries mined and showed that the information extracted from the logs was meaningful.

We also proposed a method for leveraging temporal information to improve search based on temporal result set reordering. Our results showed significant improvements could be achieved for both a general set of web queries as well as a set of ACM conference name queries using the approach. Our evaluation also showed that temporal queries are generally more difficult than non-temporal queries and that temporal reordering is the most effective for temporally unambiguous queries.

This work only scratches the surface of temporally biased queries. There are many possible directions for future work. One potentially fruitful direction includes using more general mining strategies to detect more subtle temporally-biased queries, such as “turkey”. It would also be valuable to develop a temporal query classifier. This would provide valuable information to the underlying ranking function which could use the information to perform temporal category-specific expansion. Finally, it may be possible to use this mined temporal information to develop enhanced user interfaces. Such interfaces could show users a distribution of years that may be related to the query and allow them to choose which time range, if any, they are interested in exploring.

References

[1] E. Amitay, D. Carmel, M. Herscovici, R. Lempel, and A. Soffer. Trend detection through temporal link analysis. *J. Am. Soc. Inf. Sci. Technol.*, 55(14):1270–1281, 2004.

[2] S. Chien and N. Immorlica. Semantic similarity between search engine queries using temporal correlation. In *WWW*, pages 2–11, 2005.

[3] H. Cui, J.-R. Wen, J.-Y. Nie, and W.-Y. Ma. Probabilistic query expansion using query logs. In *Proc 11th Intl. Conf. on World Wide Web*, pages 325–332, 2002.

[4] W. Dakka, L. Gravano, and P. Ipeirotis. Answering general time-sensitive queries. In *Proc. 17th Intl. Conf. on Information and Knowledge Management*, 2008.

[5] H. Daume and E. Brill. Web search intent induction via automatic query reformulation. In *HLT/NAACL*, pages 49–52, 2004.

[6] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *Proc. 27th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 18–24, New York, NY, USA, 2004. ACM.

[7] S. Dumais, E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. C. Robbins. Stuff i’ve seen: a system for personal information retrieval and re-use. In *Proc. 26th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 72–79, 2003.

[8] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.

[9] M. Jiang, A. Shlomo, C. Abdur, and S. Kush. Explaining temporal variations in query volume. In *Proc. 2nd Intl. Conf. on Advanced Data Mining and Applications*, pages 485–492, 2006.

[10] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *WWW*, pages 387–396, 2006.

[11] J. Kleinberg. Bursty and hierarchical structure in streams. In *Proc 8th Ann. Intl. ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pages 91–101, 2002.

[12] X. Li and W. B. Croft. Time-based language models. In *Proc. 12th Intl. Conf. on Information and Knowledge Management*, pages 469–475. ACM, 2003.

[13] B. Liu, R. Jones, and K. L. Klinkner. Measuring the meaning in time series clustering of text search queries. In *CIKM*, pages 836–837, 2006.

[14] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *Proc. 15th Intl. Conf. on World Wide Web*, pages 533–542. ACM, 2006.

[15] D. Metzler, R. Jones, F. Peng, and R. Zhang. Improving search relevance for implicitly temporal queries. In *Proc. 32nd Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2009.

[16] P. Ogilvie and J. Callan. Combining document representations for known-item search. In *Proc. 26th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 143–150, 2003.

[17] F. Peng, N. Ahmed, X. Li, and Y. Lu. Context sensitive stemming for web search. In *Proc. 30th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 639–646, 2007.

[18] B. Piwowarski and H. Zaragoza. Predictive user click models based on click-through history. In *Proc 16th Intl. Conf. on Information and Knowledge Management*, pages 175–182, 2007.

[19] S. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In *Proc. 13th Intl. Conf. on Information and Knowledge Management*, pages 42–49, 2004.

[20] J. J. Rocchio. *Relevance Feedback in Information Retrieval*, pages 313–323. Prentice-Hall, 1971.

[21] D. Shen. *Learning-based Web Query Understanding*. PhD thesis, Hong Kong University of Science and Technology (HKUSC), 2007.

[22] B. Tan, X. Shen, and C. Zhai. Mining long-term search history to improve search accuracy. In *Proc. 12th Ann. Intl. ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, pages 718–723, 2006.

[23] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *Proc. 28th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 449–456, 2005.