The University of Illinois' Graduate School of Library and Information Science at TREC 2011

Miles Efron, Adam Kehoe, Peter Organisciak, Sunah Suh
501 E. Daniel St., Champaign, IL 61820

1 Introduction

The University of Illinois' Graduate School of Library and Information Science (GSLIS) participated in TREC's microblog track this year–the track's first iteration. In keeping with the foundational status of the track, our goals were chosen to emphasize the role of a single factor in microblog IR: time. As such, we adhered to the strictest guidelines of the task description, using only real-time information available in the microblog corpus to inform retrieval. Our innovation involved assessing the extent to which (and the way in which) temporal factors can improve microblog search effectiveness.

2 Experimental Data

The TREC 2011 microblog collection consists of a corpus of microblog posts made available by the microblogging service Twitter[1]. Track organizers created 50 test topics and accumulated relevance judgments in a fashion similar to the standard TREC pooling method. Details of this process are available in the track overview paper.

Instead of distributing the microblog corpus via physical media or a direct download, the posts (known as "tweets") comprising the corpus were made available to participants in a two-stage process. Organizers defined the scope of the corpus by enumerating a set of approximately 16M unique tweet ID numbers. Participants downloaded these ID's, along with software that allowed them to fetch the tweets themselves directly from Twitter.

This process bears mentioning because the download process offered participants two methods of retrieving tweets. These methods yielded slightly different corpora. Twitter offers an API whose users can request a particular tweet by its ID number. Such an API call delivers not only the text of the tweet, but several pieces of metadata (mostly giving statistics regarding the tweet author) are also included. These data are delivered to the client in JSON format. However, the Twitter API enforces a rate limit that makes

[1] `http://twitter.com`

downloading the entire corpus prohibitively slow. Some participants had previous arrangements with Twitter giving them "whitelist" access to much API calls. We were among these groups. However, many participants did not have access to this resource, so we used the more restricted HTML mode of access to make our results comparable with a lowest common denominator among track participants. The HTML data lacked most metadata found in the JSON representation. Additionally some participants had difficulty acquiring particular tweets via the HTML method, yielding corpora with fewer documents than those obtained from the API. Our corpus contained 15,653,612 tweets, each containing: the user name of the author, the time at which the tweet was posted, and the tweet text itself.

3 Base System

For core indexing and retrieval we used the Indri search engine and API[2]. Our baseline approach was the simple query likelihood retrieval model. We used Jelinek-Mercer smoothing with smoothing parameter $\lambda = 0.4$. Thus, given a query Q and a document D, we derive a score according to:

$$Pr(Q|D) \cdot Pr(D) = \prod_{i=1}^{n(Q)} Pr(q_i|D)Pr(D) \tag{1}$$

where the language model probabilities are estimated by:

$$\hat{Pr}(q_i|D) = \lambda \frac{n(q_i, D)}{n(D)} + (1 - \lambda) \frac{n(q_i, C)}{n(C)} \tag{2}$$

where $n(q_i, D)$ is the number of times word q_i occurs in D, $n(D)$ is the length of D, $n(q_i, C)$ is the frequency of q_i in the collection, and $n(C)$ is the total number of word tokens in the collection. Typically we take the prior probability $Pr(D)$ to be uniform, though we describe one experiment below using non-uniform priors.

Very little pre-processing was used in our experiments. We did not stem documents. We did create a stoplist of 133 terms. A few of these terms were familiar stopwords, which we included in the stoplist to improve proposed document expansion models. But most of these words were unique to the Twitter environment. For example, we removed words such as *fb, ff, tinyurl* and *twitpic*. Again, these were removed to reduce their influence during document expansion.

To improve retrieval we made an effort to remove non-English tweets from results. We excluded all tweets that contained more than 4 characters with encoding greater than 255 to remove non-western alphabets. We also applied a very crude filter for foreign language tweets by defining a set of 132 words that are very common in languages including French, Spanish and German. Any tweet containing one of these putatively foreign words was

[2] `http://lemurproject.org`

excluded during retrieval. It is important to stress that this filtering was accomplished via a list of words that we created based on our own knowledge. We believe that this does not constitute external evidence as described below (track organizers stated that stoplists were not external evidence, and we believe that our list of foreign words is analogous to a stoplist).

4 TREC 2011 Microblog Task

A full description of the task completed by participants in the microblog track is available in the track's overview paper. We offer only a brief synopsis here. The task involved finding tweets that were relevant to a keyword query issued at a particular time. The track corpus spanned a two-week period of Twitter activity and each test query had an associated time-stamp. Thus the scenario imagined a user issuing a keyword query at time t. Track participants only retrieved those documents written prior to t. Unlike many ad hoc IR scenarios, the task did not involve relevance-based ranking. Instead, systems retrieved a set of 30 documents that they deemed most relevant. The official performance metric was precision at 30 (P30).

4.1 External and Future Evidence

Track organizers classified runs into categories based on the types of evidence they used. Evidence could consist of:

- *Internal evidence:* Data available within the corpus proper.
- *External evidence:* Information taken from sources not present in the downloaded corpus. For instance, many tweets contain links to URLs. Text acquired from linked URLs constitutes external evidence. Interestingly, many links on Twitter are shortened. Organizers defined the unshortened (i.e. resolved) text of these URLs as external evidence.
- *Future evidence:* The microblog retrieval task was inherently temporal, with each query "taking place" at a given time. Any evidence that would not have been available at query time was considered future evidence.

A surprising result of the track's evidence typology is that using corpus-level statistics requires care if we wish to avoid using future evidence. For example, smoothing language models with probabilities based on the entire corpus relies on evidence that accumulated after any of the supplied queries. To adhere to the track's strict guidelines, any smoothing must rely on word counts observed no later than query time.

All teams were required to submit at least one run that used only internal evidence. Among our stated goals was to adhere to the track's strictest guidelines in efforts to make

the effect of our proposed innovations as clear as possible. Therefore, all of our runs used only internal evidence. To remove the influence of future evidence on document smoothing we employed the following approach. For a given query Q issued at time t we retrieved $n = 2000$ documents using Indri (admittedly, with corpus-level statistics informing smoothing via linear interpolation). We then calculated background probabilities for each query word at time t using the Indri query language. The 2000 retrieved documents were then re-ranked by smoothing based on the newly estimated background model. Only the top 30 of these re-ranked documents were submitted. While it is the case that word counts observed after time t influenced the set of 2000 initially retrieved documents, the brevity of the test topics makes it unlikely that the final 30 retrieved documents were influenced by this effect. Thus we believe that our runs did not rely on future evidence.

5 Submitted Runs: Temporally Informed Microblog Search

The temporal nature of this year's track description speaks to the strong role that time plays in microblog IR. We hypothesized that relevance in this context would have an inherently temporal dimension and that capitalizing on this would improve retrieval effectiveness. Specifically, we made use of that fact that each document D has a time-stamp t_D indicating when it was posted to Twitter. Likewise, each query has a time-stamp t_Q indicating when it was issued. Our working hypothesis was that this temporal information could improve effectiveness over the simple lexical model.

When accounting for time in retrieval, we calculated a document's temporal information t_D as the amount of time that elapsed since the tweet was published and the time the query was issued, measured in fractions of days. Thus t_D is how old D is.

We tested three methods of capitalizing on recency in retrieval:

- *Temporal Priors:* Following Li and Croft [4], promote newer documents by assuming that $Pr(D)$ follows an exponential distribution on document age. [No official runs submitted.]
- *Temporal Smoothing:* Roughly following Efron and Golovchinsky [1], smooth language models of older documents more aggressively than models for recently published documents. [Runs `gus` and `gustc`.]
- *Temporal Profiles:* This was to our knowledge a novel approach to temporal IR. We describe it in depth below. [Run `gut`.]

Of these three methods, two have been described in previous work, while a third is to our knowledge novel.

5.1 Temporal Document Priors

A standard approach to letting recency influence retrieval in the language modeling framework is to introduce document priors based on time [4]. The standard query likelihood model can be augmented as follows:

$$Pr(D|Q) \propto Pr(Q|D)Pr(D|t_D) \tag{3}$$

where $Pr(D|t_D)$ can be understood as the probability that D is of general interest given that it was published at time t_D. Li and Croft use the exponential distribution with rate parameter r for this prior, giving $Pr(D|t_D) = r \exp[-r\, t_d]$. Based on prior published literature, we set $r = 0.01$.

5.2 Temporal Document Smoothing

We submitted two runs that used temporally informed language model smoothing, as in [1]. These runs, `gus` and `gustc` smooth language models more aggressively for older documents, lending retrieval a document age penalty. We use the following quantity to smooth language models:

$$\lambda_D = 0.5 \cdot (0.4 \cdot \frac{t_D}{t_{max}}) + 0.5 * 0.4 \tag{4}$$

where t_D is the age of document D as described above, and t_{max} is the time-stamp of the oldest document in the collection. Eq. 4 mixes a temporal component with our standard smoothing value 0.4 using equal weights.

5.3 Tweets as Queries: Retrieval-based Evidence

Our most successful submitted run, `gut`, was based on a novel approach to considering time in IR. This approach improved retrieval effectiveness significantly. The approach uses a two-stage process. First we perform a standard ad hoc retrieval. The second phase involves re-ranking retrieved documents based on their observed characteristics. In the case of our submitted run `gut` the observed characteristic was temporal, but this need not be the case.

The evidence that we used for document re-ranking was obtained by submitting a given document D as a pseudo-query against the collection. This yielded a retrieved set of tweets $R = r_1, r_2, \ldots, R_k$ where we set $k = 50$. We call the documents retrieved by using D as a pseudo-query the "retrieved set" for D. For instance, consider document 29983478363717633 from the microblog corpus:

```
    [BBC News] Major cuts to BBC World Service:  BBC World Service is
to close five of its language services, with th...  http://bbc.in/e2vlpX
```

This document was retrieved by the topic `MB001`, which we call Q. After processing this tweet according to the specifications described in Section 3 we have the pseudo-query:

```
bbc news major cuts to bbc world service bbc world service is to
close five of its language services with th bbc in e2vlpX
```

We ran this pseudo-query, obtaining a retrieved set of 50 documents that are putatively related to the original document 29983478363717633. During retrieval, we derive a retrieved set for each document among the top 100 tweets returned for the original query Q. We then re-ranked these 100 documents based on statistics calculated from their retrieved sets, finally returning the top 30 documents. We speculated that the retrieved set of each ranked document D lends additional information to the problem of estimating the relevance of D to Q.

The motivation for using pseudo-queries and their retrieved sets was twofold:

1. Tweets are typically about only one topic.

2. High-quality tweets are likely to evince strong topicality, while trivial, unexpressive tweets will have a less clear topic.

Points 1 and 2 suggest that the retrieved set of a relevant document is likely to have qualities that differ from non-relevant documents' retrieved sets. In particular, we focused on the time-stamps of documents. We hypothesized that the time-stamps of documents in the retrieved set for a relevant document would share a distribution similar to the distribution of time-stamps found among the documents retrieved for the initial query. On the other hand, non-relevant documents' retrieved sets will show a distribution of time-stamps that differs from Q's.

We refer to the time-stamps of document D's retrieved set as the "temporal profile" of D[3]. Specifically, for a document D, the temporal profile T_D is the times of documents retrieved by submitting D as a pseudo-query. In other words, T_D consists of the ages of the first 50 documents retrieved by using D as a pseudo-query.

Figure 1 schematizes the information that underlies our approach. The figure plots kernel density estimates obtained from the result sets from three retrievals: the original text of TREC topic MB001, a document that is relevant to MB001, and a non-relevant document that was retrieved by our system for MB001. The x-axis of the figure shows document age (measured from the end of the microblog corpus timespan). The black line plots the empirical density of the time-stamps retrieved by the original query. We can see that a large number of retrieved documents were written about two days "ago."

The blue and red lines show the densities of time-stamps observed when we submit a relevant and a non-relevant document as a pseudo-query, respectively–i.e. each document's temporal profile.

Two results are evident in Figure 1. First, the temporal profile of the relevant document allocates much of its probability mass in the same region as the query distribution's mode. On the other hand, the non-relevant document's temporal profile diverges widely from the

[3]The term *temporal profile* was coined by Jones and Diaz [2].

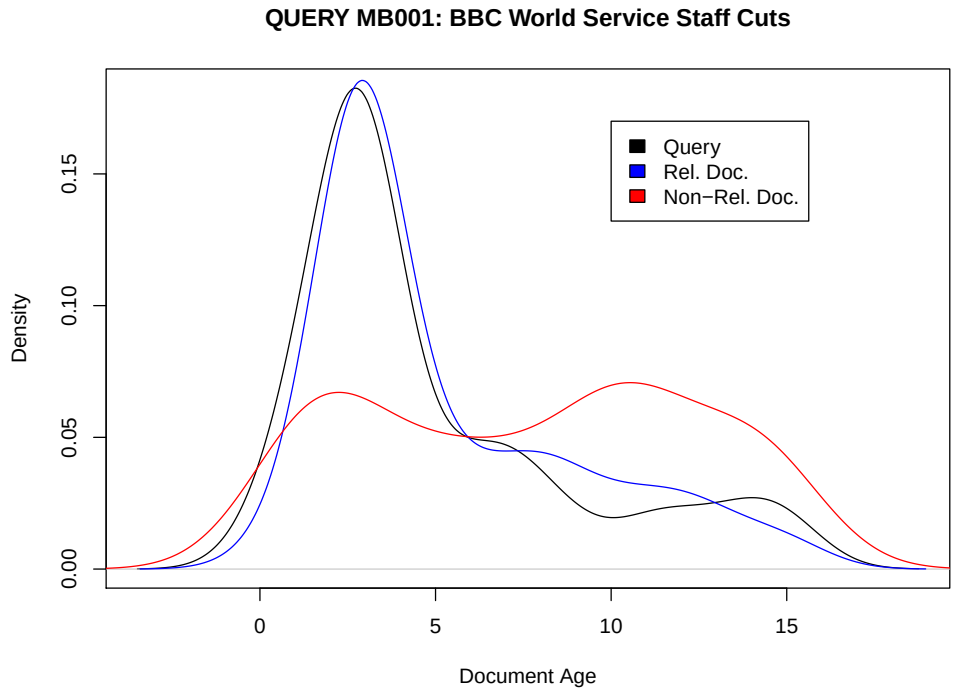

Figure 1: **Distribution of Time-Stamps of Retrieved Documents for MB001.** Kernel density plots showing the distribution of time-stamps among documents retrieved by the topic MB001, the pseudo-query for a relevant document, and the pseudo-query for a non-relevant document.

query's temporal profile. Putting it more concretely, the density of the relevant document's temporal profile has a smaller Kullback-Leibler divergence from the query profile than the non-relevant document's profile does.

Secondly, both the query and the relevant document's densities show a stark mode near $x = 2$. In contrast, the non-relevant document's temporal profile allocates probability mass more uniformly over the domain of x.

These two observations speak to points 1 and 2 listed above. Each of them influenced our retrieval approach as described in the next subsection.

5.3.1 Use of Temporal Profiles

We submitted one run that made use of temporal profiles, `gut`. For this run, a document's final score is given by:

$$s(Q, D) = \log Pr(Q|D) + \phi(T_Q, T_D) \tag{5}$$

where $\phi(T_Q, T_D)$ is:

$$\phi(T_Q, T_D) = \log(\frac{\hat{m}_{T_Q}}{\hat{m}_{T_D}}) \tag{6}$$

where $\hat{m}_{T_Q}$ is the sample mean of the time-stamps of the documents retrieved by Q, $\hat{m}_{T_D}$ is the sample mean of the time-stamps retrieved by the pseudo-query for D, and $\hat{\sigma}_{T_Q}$ and $\hat{m}_{T_Q}$ are the corresponding sample standard deviations.

The first factor in Eq. 6 promotes newer documents. Older documents are penalized, though the penalty is tempered if the query itself shows weak preference for retrieving recent documents. The second factor addresses the extent to which a temporal profile evinces temporal coherence. If the query concerns a particular time, we expect that the standard deviation of its temporal profile will be small (centered around a mode as in Figure 1). Likewise, if a document relates to a particular, time-bound event, we expect its temporal profile to have a small standard deviation.

For the sake of expediency this run abandons the formalism of the language modeling approach. Initially our hope was to capture the relationship between the temporal profiles of a query and a document by assessing the probability that the document temporal profile was generated by the same distribution that generated the query's temporal profile. However, measuring this relationship based on estimated densities proved to be very noisy on training queries that we created, and so we opted for the less elegant but simpler approach shown here.

Also, the approach shown has an implicit assumption of normality among temporal profiles. Use of the mean and standard deviation in Eq. 6 suggests that we expect temporal profiles to be centered around their means. Figure 1 suggests that this is wrong. In future work we will address this shortcoming.

6 An Aside: Semantic Profiles

The previous section introduced methods for bringing time to bear on IR using documents retrieved by using tweets as pseudo-queries. However, we could just as easily use pseudo-query results for other purposes. This section briefly proposes one such approach in which we rely on language models estimated from the a document's retrieved set to improve our assessment of query-document relation. For a given document D with a retrieved set $R = r_1, \ldots, r_k$, we induce a smoothed language model for R using Eq. 2, substituting word counts in the document for word counts in R at large. This allows us to estimate $Pr(Q|R)$ for a query Q. In one approach which we did not submit, we multiply Eq. 1 by this quantity to achieve a final document score. This is similar to building a relevance model for each document [3]. We offer this description to demonstrate that evidence gleaned from pseudo-queries could have non-temporal applications, calling the induced model R a document's "semantic profile."

7 Empirical Results

For our official runs, we retrieved only 30 documents per query since the track's official effectiveness measure was P30. Median P30 (using all judged queries) was 0.2592. Our official results all exceeded the median: temporal smoothing (`gus`=0.2973 and `gust`=0.3027[4], and temporal profiles (`gut`=0.3218).

For a more complete analysis, we computed a baseline using simple query likelihood retrieval (with no future evidence). We also re-ran each experimental condition, returning 100 tweets per query that were derived by re-ranking 300 retrieved documents. Results of these runs are shown in Table 1.

While the more established methods of dealing with temporal factors in IR (exponential priors and temporal smoothing) have mixed success, the approach using temporal profiles gave statistically significant improvements on five effectiveness metrics over the query likelihood baseline. Additional information–semantic profiles–improved retrieval at statistically significant levels for several measures. These results suggest that supplementing lexical information in microblog IR holds promise for improving retrieval.

8 Conclusion and Future Directions

An important next step in the work reported here is to assess the reason that temporal profiles improved retrieval effectiveness in our results. The less successful temporal approaches are based strictly on recency, whereas temporal profiles are a more general way of treating time. This may explain their utility. However, it may also be the case that temporal profiles improved retrieval by acting as de facto document priors. The standard

[4]The run `gust` suffered from a bug that made it not substantively different from `gus`

Table 1: **Summary Statistics for Retrieval Runs (non-official).** Mean average precision, R-precision, normalized discounted cumulative gain, precision at 10 and precision at 30 observed using a baseline query likelihood model and four experimental conditions. A + indicates $p < 0.05$ on a randomization test, and ++ indicates $p < 0.01$. Numbers in parentheses are the percent improvement (or decline, indicated with a − sign) over the query likelihood baseline.

Condition	MAP	Rprec	NDCG	P10	P30
Baseline	0.185	0.272	0.350	0.396	0.307
Exp. Priors	0.183 (-0.76)	0.271 (-0.66)	0.351 (0.40)	0.388 (-2.05)	0.310 (1.0)
TSQL	0.189 (2.00)	0.276 (1.18)	0.365 (4.29+)	0.371 (-6.19)	0.303 (-1.30)
Temp. Profiles	0.197 (6.27+)	0.286 (5.03+)	0.366 (4.55++)	0.449 (13.41++)	0.322 (5.0+)
Sem. Profiles	0.194 (5.08)	0.284 (4.37+)	0.365 (4.40+)	0.398 (0.53)	0.318 (3.58+)

deviation of temporal profiles informed our ranking, and we hypothesize that tweets whose profiles have large variance are likely to be of little interest in general. In future work we will pursue this hypothesis.

In other work, we plan to integrate information obtained from documents' pseudo-queries into retrieval in a probabilistically sound fashion. This work used pseudo-queries in an ad hoc way, importing their temporal and semantic information without proper probabilistic motivation. However, using semantic profiles for document smoothing presents an obvious way to bring their information to bear on IR. Our current work addresses these challenges.

References

[1] M. Efron and G. Golovchinsky. Estimation methods for ranking recent information. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, SIGIR '11, pages 495–504, New York, NY, USA, 2011. ACM.

[2] R. Jones and F. Diaz. Temporal profiles of queries. *ACM Transactions on Information Systems*, 25(3):14, 2007.

[3] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, New York, NY, USA, 2001. ACM.

[4] X. Li and W. B. Croft. Time-based language models. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 469–475, New York, NY, USA, 2003. ACM.