NovaSearch at TREC 2013 Microblog Track: Experiments with reranking using Wikipedia

Flávio Martins, André Mourão and João Magalhães

Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
Caparica, Portugal
flaviomartins@acm.org, a.mourao@campus.fct.unl.pt,
jm.magalhaes@fct.unl.pt

Abstract. Users engaged in microblogging services and social-media apps contribute to multiple real-time text streams which amass large volumes of messages often sparked by events reported in newswire and in other media. We explore the use of external sources to detect topic popularity surges and improve microblog search performance using time-based language models [3]. The major novelty concerns the analysis that explores Wikipedia page view streams to find topic interest spikes. We obtained promising initial results when using evidence from Wikipedia for temporal reranking with the Tweets2013 dataset.

1 Introduction

Search over real-time stream data such as the messages generated in social-media networks (e.g. Twitter) is a killer feature of these platforms. Search functionality in these online services should favour events that are occurring now. What is occurring now is discussed in many different places in the Web at the same time. We can build on top of this observation by correlating data from multiple sources to learn more about the events detected. There are two major challenges in this approach: the first is how to assert the veracity of these information sources; the second challenge concerns the analysis across different streams of data.

Wikipedia is used by millions[1] as a reliable and dependable source of insight in various subject matters. It is highly regarded as one of the most trustworthy encyclopedias. Nonetheless, Wikipedia is far from being just a set of static articles painstakingly collected over a period of years. Wikipedia is pretty much alive and evolves in real-time with the edits of its users. Similarly to what happens in social-media services, major events inspire the interest of the users towards related articles which, in turn, can also show an elevated number of edits and page views near the dates of an event. Thus, we propose to explore Wikipedia streams of data to improve search in microblogs.

[1] http://www.comscore.com/Insights/Blog/comScore_Releases_Top_50_US_Multi-Platform_Properties_for_September_2013

2 Related work

Dakka et al. [1] discusses how to deal with time-sensitive queries. It argues that queries that favor recency are just a subset of these. Therefore, the authors propose to identify the time intervals of interest for each query and to integrate this information in the scoring scheme using a number of different techniques. They evaluated their framework on news article datasets (including TREC) and with a collection of web data annotated by Mechanical Turk workers.

One approach proposed in Peetz et al. [5], leverages the temporal distribution of the documents initially retrieved. The authors hypothesize that the temporal distribution of time-sensitive document collections exhibit bursts and that documents in these bursts are more informative. Therefore, they first identify high quality documents sampled from bursty periods and then update the query model with informative terms from these documents.

2.1 Estimation methods for ranking recency

Efron and Golovchinsky [2] explores estimation methods to promote recent tweets. One of the approaches is time-based language models, proposed in Li and Croft [3], which applies time-based exponential priors to the score of the documents. The exponential prior proposed in [2, 3] for recency reranking is given by formula (1). The formula discounts the score for each tweet according to the age of the tweet in relation to the time of the query (e.g. number of days). McCreadie et al. [4] also used a variation of formula (1), in the 2011 TREC Microblog task to promote tweets in the first 6 hours.

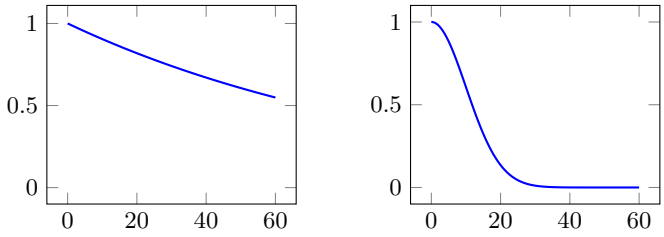
Li and Croft [3] found that the parameter λ is query-specific in their experiments. Therefore, if a fixed value of λ is used for all queries, retrieval performance can improve in some queries but can also deteriorate the performance of others. In both papers [2, 3], $\lambda = 0.01$ provides the best average performance, even though different datasets were used.

$$Texp(score, age) = score \cdot \lambda \exp[-\lambda age] \tag{1}$$

3 Proposed approach

Our approach is based on the correlation of multiple information streams that carry different types of information about the same topic. We hypothesize that topics that burst on Twitter can be highly correlated with a higher page view value and higher editing frequency of related Wikipedia pages. These pages can describe the topic itself, a related event, the entities involved, etc. We use formula (2) to perform temporal reranking using the query time, as well as using time points extracted from Wikipedia evidence. This formula aims to penalize more strongly documents that are temporally far from the time of the query.

$$Texp2(score, age) = score \cdot 0.01 \exp[-0.005 age^2] \tag{2}$$



(a) Texp (b) Texp2

Fig. 1. Temporal reranking formulas

In figure 1, we plotted the *Texp* and *Texp2* functions over the domain $(0 : 60)$ (60 days when measuring *age* in days). Notice that we normalized the functions to the interval $(0 : 1)$ to facilitate the visualization. In figure 1a, we see that the *Texp* formula generates a very soft decay. With the *Texp2* formula, we tried to make the decay in the first few days smooth so that the days next to the query time would not be penalized too much. Finally, we chose the parameters so that it would decay to 0 in about 20 days as can be seen in figure 1b.

4 Experimental setup

In this real-time ad-hoc search task, the user wishes to search for the most recent and relevant posts. The task can be summarized as: at time t, find tweets about topic X. Therefore, systems should favor relevant and highly informative tweets about the query topic posted before the query time. Due to the nature of microblogs, it is likely that relevance has a temporal dimension. That is, relevant tweets are likely to have been published nearer to the time of the query. Therefore, systems should also take into account the temporality of the tweets.

This year the track ran in a Track-as-a-Service model (to adhere to Twitter's terms of service), where participants do not have access to the whole collection; they have access to the tweets by issuing queries to a search API running in a remote server administered by the track organizers. We therefore experimented with methods to rerank the ranked list of tweets returned using the search API.

Tweets2013 corpus is the new official corpus created for the TREC 2013 microblog track. This collection consists of approximately 240 million tweets (statuses), collected via the Twitter streaming API by crawling the public stream sample over a two-month period: 1 February, 2013 - 31 March, 2013 (inclusive). NIST created 60 topics based on this corpus each representing a information need at a specific point in time. The assessors judged the relevance of the tweet but also considered the relevance of any URLs linked from the tweet. All assessments were conducted by NIST assessors on a three-point scale of "informativeness": not relevant, relevant and highly relevant. The primary evaluation measure for this year is MAP, however precision at rank 30 cutoff and R-prec are also reported.

5 Experimental results

Assessors consider tweets not written in the English language as not relevant. Moreover, a retweet is generally considered not revelant. It is only relevant if it is an *RT-style*, where the author added highly informative text to the original tweet. Therefore, our baseline run combines a language filter and a retweet filter.

Language filter. We use a language filter to remove tweets that are not written in English. Before passing the tweet to the language identification library, we put the text through some processing steps. We use twitter-text-java[2] to detect and remove URLs, @mentions, lists and #hashtags from the tweet. We remove other common patterns used pervasively in Twitter, regardless of the language that could confuse the language detector: RT, \via and its variants and other prevalent microsyntax such as \by and \cc.

To detect the language of the tweets we used the library language-detection[3] but relaxed the identification. We aim for a low number of false negatives: tweets that are in fact written in English but are not detected. The detector is used in a way that English must only be one of the probable languages, and doesn't have to be the language with the highest probability.

Retweet filter. The retweet filter simply discards retweets from the results. We filtered *Twitter-style* retweets using the available tweet metadata. We further filter out *RT-style retweets* that start with the word "RT", since it is standard for users to add commentary before of the retweeted text and not after. Therefore, these are tweets that will be most likely judged as not relevant by assessors.

NOVAsearch00: Language and retweet filters combination. In this run we remove retweets and non-English tweets with the filters described above. This run shows an improvement from the QL Baseline of 20.6% in P30 and MAP, while R-prec improves 16.7% (see Table 2).

NOVAsearch01: Temporal reranking using query time. The ranked list obtained using our baseline (NOVAsearch00) can be improved by reranking using only the temporal information available at query time (the actual time when the query was issued). For reranking, we selected the *Texp2* function that gave us the best results for P30 in the 2011 and 2012 queries with the Tweets2011 corpus. However, this was not the case with the new dataset and this function performed poorly against *Texp* (see Table 2). The *Texp2* function decays too fast and penalizes posts that are perhaps too recent for the 2 month time span of the newer evaluation corpus (see Figure 1b).

[2] https://github.com/twitter/twitter-text-java
[3] https://code.google.com/p/language-detection/

Table 1. Topics and Wikipedia page matching: shows the query date and the Wikipedia page views peak date.

Topic	Query	Wikipedia Page	Q-Time	Wiki-Time
MB115	memories of Mr. Rogers	Fred Rogers	Mar 30	Mar 20
MB127	Hagel nomination filibustered	Chuck Hagel	Mar 5	Feb 27
MB132	asteroid hits Russia	Asteroid impact avoidance	Feb 20	Feb 15
MB133	cruise ship safety	Cruise ship	Mar 15	Mar 15
MB164	Lindsey Vonn sidelined	Super-G	Mar 30	Mar 19

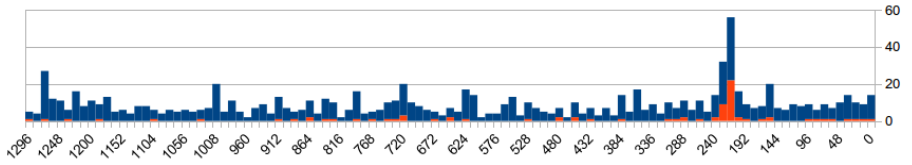
NOVAsearch02: Temporal reranking using Wikipedia page view data. We hypothesized that a topic bursts simultaneously in Twitter and Wikipedia. More specifically, we hypothesize that Wikipedia pages related to a topic show a surge of page views when a topic bursts on Twitter. Therefore, we leveraged page view log data from Wikipedia to estimate when a topic might have burst. To select a related page for each topic, we simply queried the Wikipedia search API with the original topic text, as a user would, and used the first hit returned in the results. We used both the query time as well as the the page views peak date from Wikipedia (limited to up to 10 days before query time) for reranking. We show the pages selected for a small number of topics in Table 1 as well as their Wikipedia page views peak date. This information can be matched visually with the plots in Figure 2.

NOVAsearch03: Document expansion with linked documents. In this run we experimented with document expansion. We processed the web pages linked from the tweet messages using Goose[4] in order to extract their titles and metadata. The representation of a tweet document was expanded in the index by adding a new field that contains the title of tweets' linked web documents. We reindexed only 10,000 tweets for each topic, due to the document limit of the search API. The results of this experiment were not encouraging.

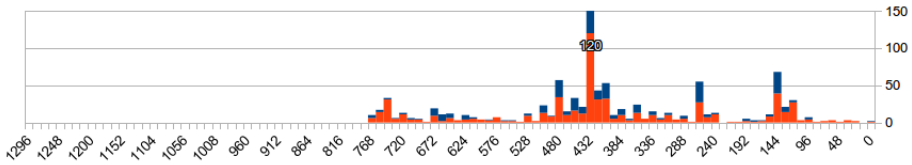
Table 2. TREC 2013 Microblog track: Real-time ad-hoc task results

	All relevant			Highly relevant		
Method	MAP	R-prec	P30	MAP	R-prec	P30
QL Baseline	0.2044	0.2699	0.3761	0.1540	0.1946	0.1900
NOVAsearch00	0.2726	0.3171	0.4711	0.1941	0.2280	0.2306
NOVAsearch01	0.2082	0.2530	0.4367	0.1607	0.2064	0.2222
NOVAsearch02	0.2239	0.2738	0.4450	0.1696	0.2166	0.2267
NOVAsearch03	0.1612	0.2092	0.3550	0.1341	0.1651	0.1911

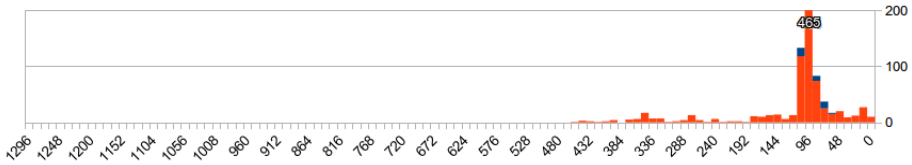
[4] https://github.com/GravityLabs/goose



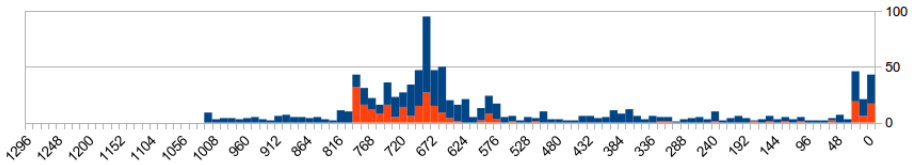
(a) Topic MB115: “memories of Mr. Rogers”



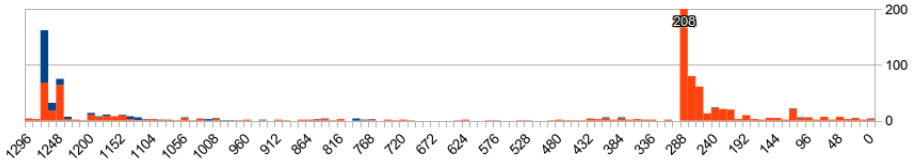
(b) Topic MB127: “Hagel nomination filibustered”



(c) Topic MB132: “asteroid hits Russia”



(d) Topic MB133: “cruise ship safety”



(e) Topic MB164: “Lindsey Vonn sidelined”

Fig. 2. Temporal distribution of retrieved tweets: The horizontal axis is a timeline indexed by hours until the query time. (Relevant documents represented in red.)

6 Summary and Future Work

Our system augments the query-likelihood model with time-based language models, using temporal exponential priors and our additional formula Texp2. With this setup, we were able to improve ranking performance by correlating the topic with Wikipedia page view data for temporal reranking.

Simple and fast techniques that seamlessly integrate the temporal aspect of the queries are desirable. In query-likelihood retrieval, the time-based language model approach is most certainly one of the most straightforward techniques that integrates the temporal aspect of queries or recency in retrieval. To this effect, we will pursue in the future cross-stream analysis methods to correlate multiple streams and discover useful information to improve microblog retrieval.

Acknowledgments

This work has been partially funded by the Portuguese National Foundation under the projects PTDC/EIA-EIA/111518/2009 and UTA-Est/MAI/0010/2009.

References

1. Dakka, W., Gravano, L., Ipeirotis, P.: Answering general time-sensitive queries. IEEE Transactions on Knowledge and Data Engineering 24(2), 220–235 (2012)
2. Efron, M., Golovchinsky, G.: Estimation methods for ranking recent information. In: SIGIR '11. p. 495504. SIGIR '11, ACM, New York, NY, USA (2011)
3. Li, X., Croft, W.B.: Time-based language models. In: CIKM '03. p. 469475. CIKM '03, ACM, New York, NY, USA (2003)
4. McCreadie, R., Macdonald, C., Santos, R., Ounis, I.: University of glasgow at TREC 2011: Experiments with terrier in crowdsourcing, microblog, and web tracks. (2011)
5. Peetz, M.H., Meij, E., Rijke, M.d.: Using temporal bursts for query modeling. Information Retrieval pp. 1–35 (July 2013)