2I1-5

A Time and Context Aware Re-ranker for Microblog Retrieval

Abu Nowshed Chy Md Zia Ullah Masaki Aono

Department of Computer Science & Engineering
Toyohashi University of Technology
Toyohashi, Aichi, Japan

Microblogs, especially Twitter, have become an integral part of our daily life for searching latest news and events information. Due to short length characteristics of tweets, only content-relevance based search result cannot satisfy user's information need. Recent work shows that considering temporal aspects in this regard improve the retrieval performance significantly. In this paper, we propose an approach that not only consider textual features but also temporal features, account related features and twitter specific features of the tweets for re-ranking the search results. A linear ranking model has been adopted to combine these features to estimate the relevance score. We conducted our experiments based on the topics and datasets of TREC Microblog track 2011 and 2012. Experimental result demonstrates the effectiveness of our approach over the baseline methods in terms of Precision@30, mean average precision (MAP), and reciprocal-precision (R-Prec) metrics.

1. Introduction

Nowadays, microblogging web sites are not only the places of maintaining social relationship but also can be used as a valuable information sources. Everyday lots of people turn into microblog sites for sharing their views, opinions, experiences, important news, and also want to get some information what is happening around the world today. Among several microblog sites, Twitter*1 is now the most popular, where lots of people posting tweets when a notable event occurs. That is why information retrieval in twitter has made a hit with a lot of complaisance. By searching twitter documents, people will find temporally relevant information such as breaking news and real time contents. For example, we can consider a situation, where a journalist prepares an investigation report about a sports scandal. So, to find out more details about the scandal, he turns to searching tweets from several different aspects, such as: the scandal's major facts, reactions from fellow athletes, commentary from analysts, and so on. Here, the journalist only wants to see the most recent tweet about the scandal; thought any search that contains the athletes name will brings up the results from several different points in time [1]. We can consider another situation, where the journalist writes a report about the impact of social media during the general election of Japan. In that case, he also search tweets in a time frame. So, for exploring such kind of search behaviour and boosting the retrieval performance in such environment, TREC was first introduced the ad-hoc search task in 2011 [2], where a user's information need had been represented by a query at a specific point in time and a set of relevant ranked tweet documents had been returned based on that query. In this paper, we propose an approach to re-rank the tweets that are retrieved using baseline method. To improve the re-ranking result, we consider textual features, temporal features, twitter specific features, and account related features. Experimental results with TREC microblog dataset shows that our approach improves the retrieval performance. The rest of this paper is structured as follows: **Section 2** describes the state-of-the-art of tweet search task while some retrieval models to comprehend consequent contents of our paper are articulated in **Section 3**. Next, we will introduce our approach in **Section 4**. **Section 5** includes experiments and evaluation to show the effectiveness of our proposed approach. And finally concluded remarks and some future directions of our work described in **Section 6**.

Contact: Masaki Aono, Toyohashi University of Technology, 1-1 Hibarigaoka, Tempaku-cho, Toyohashi, Aichi, Japan. 0532-44-6764, aono@tut.jp

*1 https://twitter.com

2. Related Work

Ad-hoc retrieval in microblog environment, such as twitter, is one of the state of the art research task in information retrieval domain, where the major goal is to return ranked tweet documents based on user's query. As microblog search queries are typically time-related, some researchers consider temporal properties (e.g., temporal variation and recency) as important factors [3] [4] for retrieving relevant tweets. Their results showed that time dimension taking into account improves the retrieval efficacy of temporal queries. Since tweets are limited to 140 characters in length and the average length of the queries in microblog is about 1.64 words, the vocabulary mismatch problem exacerbates the difficulty of query term matching during the retrieval [5]. Modern and representative retrieval models including Inverse Document Frequency (IDF), Okapi BM25, Language Model, Vector Space Model, Probability Ranking Principle (PRP), and etc. are used by several researchers [6] [7] to improve the retrieval efficacy. Like content based and temporal feature, other twitter related features are also proposed by several researchers [8] [7]. Moreover, Jaeho Choi et al. develop a user behavior based quality model to indicate the correlation between tweet document informativeness and relevance judgments [9].

3. Retrieval Models

In the following sections, we briefly describe some classical information retrieval model:

3.1 Inverse Document Frequency (IDF)

The Inverse Document Frequency is a measure of the general importance of a query term.

$$IDF_{t,D} = \log_2 \left(\frac{|D|}{|d \in D : t \in D|} \right)$$

Where, $|D|$ is the total number of documents in the corpus and $|d \in D : t \in D|$ is the number of documents where the query term t appears.

3.2 Language Model

In the language modeling approach, each document in the corpus generated by a probability distribution over the terms in the vocabulary. For a given query $Q = q_1, q_2, ..., q_n$, retrieved documents $D = d_1, d_2, ..., d_n$ are then generated based on the probability that the document's language model generate.

$$P(D|Q) \ \alpha \ P(Q, D).P(D)$$

Assuming uniform priors over documents and term independence:

$$P(Q|D) \ = \ \prod_{i=1}^{|Q|} P(w_i|D)$$

Where, $|Q|$ is the number of words in the query. Using multinomial language models, the maximum likelihood estimator will be:

$$P_{ml}(w|D) \ = \ \frac{n(w|D)}{|D|}$$

To improve the accuracy of the maximum likelihood estimator smoothing is performed. The form of a Dirichlet-smoothed language model is:

$$P(w|D) \ = \ \frac{|D|}{|D| + \mu} P_{ml}(w|D) + \frac{\mu}{|D| + \mu} P(w|C)$$

Where, $P(w|C)$ is the collections language model.

3.3 Okapi BM25 Model

Okapi BM25 model is a bag-of-words retrieval function that measures the content relevancy between Query Q_0 and tweet T. The standard BM25 weighting function is formulated as:

$$\sum_{q_i \in Q_0} \frac{IDF(q_i) * TF(q_i, T) * (k_1 + 1)}{TF(q_i, T) + k_1 * (1 - b + b * \frac{Length(T)}{Avg_{Length}})}$$

Where $IDF(q_i)$ is inverse document frequency, $TF(q_i, T)$ is the frequency of term q_i in tweet T, $Length(T)$ is the length of tweet T and Avg_{Length} stands for average length of tweet in the corpus.

3.4 Divergence From Randomness (DFR) Model

Divergence From Randomness (DFR) is a probabilistic approach which can be used as a query-dependent ranking model. DFR models build upon the intuition that the more the content of a document diverges from a random distribution, the more informative the document is. The standard DFR weighting function is formulated as:

$$DFR_{t,d} = \frac{tf_{t,d}(1 - \frac{tf_{t,d}}{l_d})^2}{tf_{t,d} + 1} \log_2(tf_{t,d} \frac{\bar{l}}{l_d tf_{t,C}}) + 0.5 \log_2(2\pi tf_{t,d}(1 - \frac{tf_{t,d}}{l_d}))$$

Where, $tf_{t,d}$ is the occurrences of a term t in a document d, $tf_{t,C}$ is the frequency of the term t in the corpus C, l_d is the length of d and $\bar{l}$ is the average length of all documents in the corpus.

4. Our Approach

The goal of our re-ranker system is to rank the tweets that are retrieved by using baseline method. In this regard, we perform a number of steps. At first, by using TREC search API, we fetched the 10,000 tweets for each topic. Next, at the preprocesssing stage, we removed the non-english tweets, retweet, and future tweets. After that, we extracted 10 features for ranking which are grouped into four different categories named as content relevance features, twitter specific features, account related features, and temporal features. Finally, after performing the feature normalization a linear ranking model is applied to combines all the feature value and estimate the relevance score. Based on the relevance score, we ranked all the tweets and take top 1000 tweet as relevant. The overview of our proposed approach depicted in Figure 1.

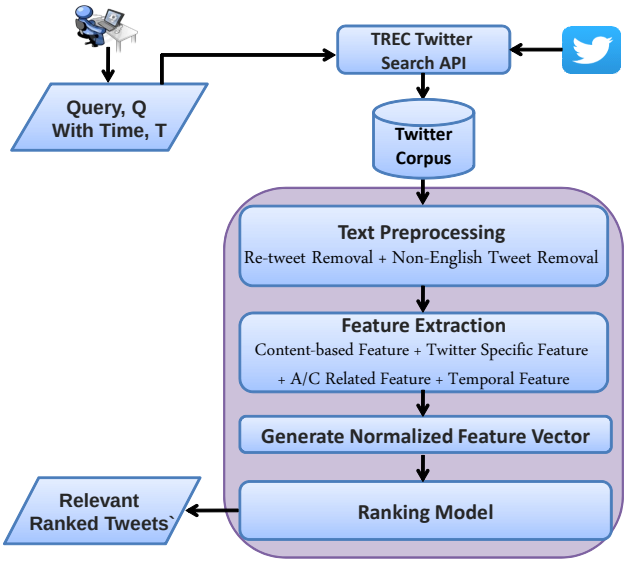

Figure 1: Tweet Re-ranker System

4.1 Dataset Collection

The official collection of TREC Microblog consists of approximately 240 million tweets. We use the official TREC Microblog Search API [10] for crawling the dataset and obtain the top 10,000 tweets for each topic as our corpus. In our experiments, we used two different TREC topic sets numbered 1-50 and 51-110, which are the official topics in the TREC 2011 and 2012 microblog track, respectively. Each topic is composed of query_id, query_text, query_time etc.; while each tweet document is composed of tweet_id, screen_name, tweet_time, tweet_text, followers_count, statuses_count, retweeted_count, and so on.

4.2 Data Preprocessing

In the preprocessing stage, initially a filtering component has been adopted to refine the crawled results based on re-tweet removal and Non-English tweet removal. Tweets that begin with the sign of RT are regarded as re-tweets and eliminated from the corpus with the consideration that they are just the copy of other tweets without any useful information. Though twitter is a multilingual microblog environment, but in our experiment Non-English tweets are judged non-relevant because all of our topics are expressed in English. To remove the Non-English tweet from the corpus, we used a Java language detection library[*2], which uses Naive-Bayesian filtering and claims over 99% precision for 53 languages. In additions, we removed the web links, Non-English characters from the tweet text. Besides these, we also removed the future tweets in some of our experiments. Tweets that are posted after the topics timestamp were treated as future tweet.

4.3 Feature Selection

We define a set of 10 features grouped into 4 different categories. The following subsections show details of these features:

4.3.1 Content Relevance Feature

Content relevance feature indicate the content relevancy between a given query and a target tweet. Here, we used four content relevant features, **Okapi BM25**[8], **Vector Space Model** [11], **Language Model** [6], and **Divergence From Randomness** [12]. As the TREC Microblog API used language model for ranking tweet, we utilize the relevance score value of the target tweet as our language model feature.

4.3.2 Twitter Specific Feature

- **URL:** URL is a binary feature that is assigned 1 if a tweet contains at least one URL and 0 otherwise.

- **#Hashtag:** Hashtag is used by users within a tweet to highlight a topic. #Hashtag is a binary feature that is assigned 1 if a tweet contains at least one URL and 0 otherwise.

- **Retweet Count:**
  In twiiter, more informative tweet that is reposted by other users without any modification is called retweet. Retweet count (RTCount) indicates the number of times a tweet is retweeted. To measure the popularity of a tweet we use an integer (RTP) between 0 and 5 (inclusive) based on the retweet count.

$$RTP = \begin{cases} 0, \text{ If RTCount} == 0 \\ 1, \text{ If RTCount} \in [1,10] \\ 2, \text{ If RTCount} \in [11,100] \\ 3, \text{ If RTCount} \in [101,1000] \\ 4, \text{ If RTCount} \in [1001,10000] \\ 5, \text{ For Other Values} \end{cases}$$

4.3.3 Account Related Feature

- **Followers Count:**
  Followers count (FCount) indicates the number of followers that the author of this status has. To measure the credibility of a tweet author we use an integer (FCP) between 0 and 5 (inclusive) based on the followers count.

$$FCP = \begin{cases} 0, \text{ If FCount} == 0 \\ 1, \text{ If FCount} \in [1,10] \\ 2, \text{ If FCount} \in [11,100] \\ 3, \text{ If FCount} \in [101,1000] \\ 4, \text{ If FCount} \in [1001,10000] \\ 5, \text{ For Other Values} \end{cases}$$

- **Status Count:**
  Status count (SCount) of a tweet indicates the number of tweets that the author posted before at the time of posting this tweet. To measure this feature we use an integer (SCP) between 0 and 5 (inclusive) based on the status count.

$$SCP = \begin{cases} 0, \text{ If SCount} == 0 \\ 1, \text{ If SCount} \in [1,10] \\ 2, \text{ If SCount} \in [11,100] \\ 3, \text{ If SCount} \in [101,1000] \\ 4, \text{ If SCount} \in [1001,10000] \\ 5, \text{ For Other Values} \end{cases}$$

4.3.4 Temporal Feature

- **Temporal Feature:**
  Based on the query time, we measure the recency score of a tweet as follows:

$$RecencyScore = \frac{1}{\sqrt{Query_{Time} - Tweet_{Time} + 1}}$$

  Where, $Query_{Time}$ denotes the time stamp of the Query and $Tweet_{Time}$ denotes the time stamp of the target tweet.

4.4 Feature Normalization

Since the range of values of different features varies widely, feature normalization technique is used to standardize the range of independent feature value. Here, we use the max-min normalization technique for rescaling the feature value within the range [0, 1].

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Where, x is the original value and x' is the normalized value.

*2 https://code.google.com/p/language-detection/

4.5 Ranking Model

To re-rank the retrieved tweets based on different features, we designed a simple linear model. This linear model combines the different features value of each tweet and estimate the relevance score value. For a given query topic (Q_{Topic}) and a tweet document (T_{Doc}), the relevance score value (rsv) can be estimated as follows:

$$rsv(Q_{Topic}, T_{Doc}) = \sum_{i=1}^{N} f_i(Q_{Topic}, T_{Doc})$$

Where, N is the number of features.

5. Experiments and Evaluation

TREC Search API provided ranking results by using Lucenes implementation of query-likelihood (LMDirichletSimilarity), which we consider our baseline. To evaluate the performance, we used three measures Precision@30, MAP, and R-Precision. Table 1 and table 2 shows the summarized results of our experiments. Re-ranking based on baseline method plus preprocessing resulted in KDE_Run1 while in KDE_Run2, we only consider the content based feature. Next, result that showed in KDE_Run3 consider the baseline method plus recency score and result based on combining all features are showed in KDE_Run4.

Table 1: TREC Microblog 2011 Dataset

Method	P@30	MAP	R-Prec
Baseline	0.3483	0.3509	0.3050
KDE_Run1	**0.4231**	**0.4099**	**0.3833**
KDE_Run2	**0.3571**	**0.3626**	**0.3265**
KDE_Run3	0.3252	**0.3710**	**0.3391**
KDE_Run4	0.1966	0.2198	0.2083

Table 2: TREC Microblog 2012 Dataset

Method	P@30	MAP	R-Prec
Baseline	0.2932	0.2354	0.1815
KDE_Run1	**0.3554**	**0.2939**	**0.2254**
KDE_Run2	0.2655	0.2318	0.1717
KDE_Run3	**0.3102**	**0.2749**	**0.2079**
KDE_Run4	0.1808	0.1629	0.1184

Our findings from the experimental results showed that the best result achieved when applying all preprocessing steps in baseline method. Considering content relevance feature and temporal feature slightly improves the retrieval performance compared with baseline for some extents. However, we did not get significant improvements while considering other features. This might be caused by corpus size or lack of applying machine learning techniques. Further investigation is still needed in this regard.

6. Conclusion and Future Direction

In this paper, we reported our preliminary experiment result based on TREC Microblog ad-hoc search task. The results showed that our proposed tweet re-ranking approach is more affected by content relevance feature and temporal feature, but is less affected by other twitter related feature.

In future, we have a plan to explore more temporal and other twitter related features deeply, which might have significant influence on tweet re-ranking task. Next, to improve the retrieval efficacy of our framework, we also have a plan to apply query expansion and document expansion along with standard machine learning techniques. Moreover, we intend to address the problem of searching result-diversification, as top-ranked documents returned by traditional retrieval functions cannot satisfy different user needs.

References

[1] J. Lin, M. Efron, Y. Wang, and G. Sherman, "Overview of the trec-2014 microblog track (notebook draft)."

[2] I. Ounis, C. Macdonald, J. Lin, and I. Soboroff, "Overview of the trec-2011 microblog track," in *Proceeddings of the 20th Text REtrieval Conference (TREC 2011)*, 2011.

[3] T. Miyanishi, K. Seki, and K. Uehara, "Combining recency and topic-dependent temporal variation for microblog search," in *Advances in Information Retrieval*. Springer, 2013, pp. 331–343.

[4] J. Lin and M. Efron, "Temporal relevance profiles for tweet search," in *SIGIR Workshop on Time-aware Information Access*. Citeseer, 2013.

[5] J. A. R. Perez, A. J. McMinn, and J. M. Jose, "University of glasgow (uog_twteam) at trec microblog."

[6] T. Miyanishi, K. Seki, and K. Uehara, "Improving pseudo-relevance feedback via tweet selection," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM, 2013, pp. 439–448.

[7] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H.-Y. Shum, "An empirical study on learning to rank of tweets," in *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 295–303.

[8] T. El-Ganainy, Z. Wei, W. Magdy, and W. Gao, "Qcri at trec 2013 microblog track," in *Proceedings of TREC*, 2013.

[9] J. Choi, W. B. Croft, and J. Y. Kim, "Quality models for microblog retrieval," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1834–1838.

[10] J. Lin and M. Efron, "Overview of the trec-2013 microblog track," in *Proceedings of TREC*, 2013.

[11] C. H. Lau, Y. Li, and D. Tjondronegoro, "Microblog retrieval using topical features and query expansion." in *TREC*. Citeseer, 2011.

[12] R. L. T. Santos, "Explicit web search result diversification," Ph.D. dissertation, University of Glasgow, 2013.