# Finding Topic-related Tweets Using Conversational Thread

Peng Cao[1,2], Shenghua Liu[1], Jinhua Gao[1,2], Huawei Shen[1], Jingyuan Li[1], Yue Liu[1], Xueqi Cheng[1]

[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100190
[2] Graduate School of Chinese Academy of Sciences, Beijing, 100190

**Abstract.** Microblog has gained more and more users around the world, the popularity of which makes information spreading in microblog the most important and influential activities on the Internet. Therefore, search in microblog is of the most significant issue for both academic and industrial world. Search in webpages has been studied for several decades, but as for microblog it is still an open and brand new question for everyone. Search in microblog is more difficult than that in traditional webpages because of the sparseness of the messages. Search functions in current microblogging services simply match microblog messages with query words, which cannot guarantee the correlation between the retrieving messages and the users' intention. We introduce the concept of conversational thread to gain more information and improve the search result in microblog. We also use SVMRank to train a model to determine the rank of relevance of the queries and messages. Through a series of experiments, we proved that our method is easy to implement, and can improve the precision up to 29% in average.

**Keywords:**Microblog, Search, Conversational Thread, SVMRank

## 1 Introduction

In the age of social web, the microblogging service as a media is prevalent, in which the posts carry massive information and spread aggressively through the Internet. Twitter, as one of the most popular microblogging services generates up to 230 million tweets[1] per day by September, 2011, according to Michael Abbott, Twitter's ex-VP of engineering. A tweet[2] differs from a post in traditional blogs in that its content is typically restricted to a very short length, typically 140 bytes or characters [1]. A tweet is "real time", the value of which drops dramatically after it has been posted for more than a day or even a few hours [2]. Microblog users read tweets in their timelines, which are posted or forwarded by their friends. Furthermore, trending topics are another important feed to help users access interesting tweets. Both reading tweets from friends and trending topics are passive ways to get information. Nevertheless,

[1] the posts named under Twitter
[2] We borrow the name "tweet" to refer the posts in a general microblogging service.

there are requirements to find information with users' intension, such as web search engines. In this paper, we study on the ad-hoc search of tweets, to find topic-related tweets according to user's descriptions, which has become one of the most popular academic and industrial interests. The Text REtrieval Conference (TREC) 2011 proposed this problem as the microblog task for the first time.

Web search known as a traditional search problem has been studied for several decades. Although a collection of classical models and algorithms on crawling, indexing, ranking and evaluation, etc. have been proposed, there are still many challenging issues, especially in searching tweets:

1. Extremely short. The length of a tweet is too short to contain enough topical features. However, those topical tweets are not independently posted in the microblogging services. Thus, considering the context of tweets to enrich its topical features is a key to find the topic-related tweets.
2. Realtime. The tweets in a microblog are posted rapidly. Furthermore the microblog users may have different intensions from the traditional web search engine. They may want to know statuses of particular topics such as some news about famous stars, the newest progress of events, etc. On the contrary, a webpage search engine mostly helps user with the navigational (give me the url of the site I want to reach) or transactional (show me sites where I can perform a certain transaction, e.g. shop, download a file, or find a map) need[3]. The freshness is more important than the relevance in microblog search. Thus the results of ad-hoc tweet searching must be organized in time-reverse order. Thus the ad-hoc searching has a definite intention to find the status of a happening event or topic, so that tweets are called statuses sometimes. Because of the importance of the freshness the ad-hoc searching, therefore, should present the results in a time-reverse order.
3. User authority. In such a social media, users in the microblogging services play a significant role in the value of tweets' content. Thus the importance of a tweet is not only decided by its content, but also the authority of its author.

This paper focuses on searching new and interesting tweets relevant to a given topic description. A query time is also given for each query, indicating the exact time of the query issuance. The search is supposed to be conducted onsite, and those tweets later than the query time should not be returned. The conversation thread is a set of tweets and their replies. It is supposed that tweets in the same conversational thread are more relevant to the same topic than other tweets outside. We introduce the concept of conversational thread as a very useful tweets context to expand the extreme short content. We propose an enhanced BM25 to determine the relevance of the tweet and the query. Furthermore we bring the tweets' freshness and activities as features. We then use SVM rank to learn the importance of those features, and our ranking results are truncated and filtered by relevance orderly. Finally those truncated results are organized in reverse-time order. In the experiments, we use precision@30 for evaluation, and our approach achieves 29% more relevant and fresh results totally.

The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 describes our main method to search and rank the search results. Section 4 describes our experiment results. We conclude in section 5.

## 2 Related work

The existing related work about microblog mainly focuses on users [4-8], information flow [9,10], and tweets' content [11-14]. [2-7] studied Twitter from the users' point of view, such as users' intentions, followers and friends, users' network, and the social features of Twitter, etc. [15] proposed a new ranking method RE(Reader Effective). [9] proposed 2 new concepts stickiness and persistence to detect the information diffusion on twitter. [10] judged the information flow among twitter users. [16] proposed some important features of tweets via PCA(primary content analyze). [11] evaluated the credibility of tweets' content via training a classifier based on message-based features, user-based features, topic-based features and propagation-based features. [12] used the instant content from Twitter to detect earthquakes timely, and collected the geographic and damage information from tweets' descriptions and users' locations as soon as possible. There were also many works focus on thread. [17] focused on the discovery of reply relationship in twitter and used simple features to train model to recover the conversational threads. [18] is the phD dissertation of the same author as [17] but gave the method and steps more detailed. [19] analyzed the statistical characteristics of Slashdot and summarized the patterns of threads in it.

Our work is to query the tweets' content to find relevant, interesting, and fresh tweets via different threads. We use the whole TREC corpus to study global features of tweets' such as content, hashtags, urls, post time, etc. Because the tweets in twitter are very short (no longer than 140 characters), there is little useful information in a single tweet and the relevant between a query and a tweet is difficult to determine. So we also introduce other tweets in the same thread to gain more information to improve the search result. We employ SVM ranking model to rank our query results. The model is trained on pair-wise labeled data.

## 3 Ad-hoc Search

The task of Microblog trec 2011 can be described as follow, given a query Q at time T, to find relevant tweets whose post times are no later than T to keep time reverse order(from newest to latest) to achieve high precision@30. A query is a simple topic described by some words.

The corpus is some specified users' tweets on twitter during Jun. 23, 2011 and Feb. 8, 2011. The whole data set is crawled from Twitter with specified seeds. After extracting from the crawled html pages, we got the tweet ID, screen name, http status code, post time, tweet content as a one-line record for each tweet. We only focus on English tweets. After preprocessing the tweets we got 5,650,490 English from 16,141,812 in total.

SVM Rank is an efficient machine learning method to rank. It uses Support vector machine (known as SVM) to rank and the kernel function should be linear. We study the distribution of each feature and take the logarithm of the user activeness and retweetness of the tweets to make them follow the power-law. Because if the distribution follows power-law, it is linear in log-log scale and it fits the SVM perfectly. Table 1 shows the features we used to train our model.

**Table 1.** Features used to train our model

| Feature name | Description |
|---|---|
| Enhanced BM25 | The content relevant |
| User activenss | The author's authority |
| Twitter Length | The number of words |
| Time of retweetness | Popularity |
| URL | Whether has urls |
| Hashtag | Hashtag relevant |
| Freshness | Resolve realtime |

### 3.1 Conversational Thread

Thread originates from Email network with obviously reply relationships. However there are also conversational threads in forums, BBS and blogs. We can also define thread in twitter similarly. A conversational thread in twitter is a rooted tree whose nodes are tweets. The children of each node(if any) is the reply tweet of it. We consider the tweets in a conversational thread to be relevant to each other and thus be more relevant to a topic than other tweets outside the tree. Thread is the implicit structure which is helpful to determine the relevance to a particular topic.

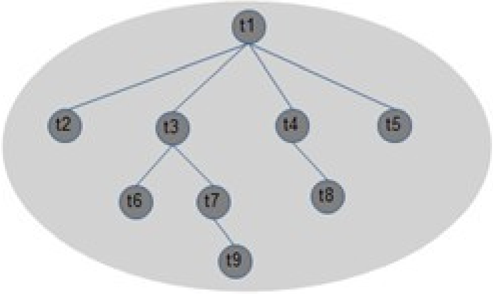

**Fig. 1.** Conversational thread

38,482 tweets in the corpus are labeled to be relevant or irrelevant to a particular topic. We crawled twitter data and get the conversational threads. We get 48,543 tweets in total to train and test our model. Figure 1 shows a typical conversation thread which has 9 tweets.

The height of a thread can be defined as the number of tweets in the longest reply-to links. Table.2 shows the height distribution of threads. It is not surprising that most of the thread is only one node. There are 38469 threads in total and about 90.34% of them have only 1 tweet. However, the average height of threads higher than 1 is about 3.80. That is to say we can have (if any) nearly 4 tweets to complement a single tweet.

**Table 2.** The height of threads

| Height | Number of threads |
|---|---|
| 1 | 34754 |
| 2 | 2172 |
| 3 | 553 |
| 4 | 298 |
| 5 | 185 |
| 6 | 124 |
| 7 | 84 |
| 8 | 62 |

### 3.2 Enhanced BM25

As for measuring the content relevance, we proposed a scoring method for tweets based on BM25.

Let Q be a query that consists of query words q1, q2, q3, etc. In the BM25 model, the score B(Ti, Q) for tweet Ti is calculated in the following way.

$$B(T_i, Q) = \sum_{j=1}^{|Q|} IDF(q_j) \cdot \frac{f(q_j, T_i) \cdot (k_1 + 1)}{f(q_j, T_i) + k_1 \cdot (1 - b + b \cdot \frac{|T_i|}{avgtl})}$$

where $q_j$ is the jth query word, |Q| is the total number of query words, avgtl is the average tweet length, and $|T_i|$ is the length of tweet $T_i$. The function $f(q_j, T_i)$ indicates the frequency of the query word $q_j$ in tweet $T_i$. The model parameters $k_1$ and b are 2.0 and 0.75 separately. IDF($q_j$) is defined as follows.

$$IDF(q_j) = \log \frac{N - n(q_j) + 0.5}{n(q_j) + 0.5},$$

where n($q_j$) is the number of tweets that contain query word $q_j$ . It is seen that the score of a tweet matching two different words is the same to that of another matching the same word twice, if the two query words have the same IDF value. Since tweets are extremely short, the word frequency actually does not count much, and matching different query words makes much more sense. Therefore, we need a way to boost such a case. The enhanced BM25 finally defines as follows with boosting where $h_j$,i is a 0-1 binary that indicates whether query word $q_j$ hits tweet i.

$$B(T_i,Q)=\left(\sum_{j=1}^{|Q|} IDF(q_j)\cdot\frac{f(q_j,T_i)\cdot(k_1+1)}{f(q_j,T_i)+k_1\cdot(1-b+b\cdot\frac{|T_i|}{avgtl})}\right)\cdot\sum_{j=1}^{|Q|} h_{i,j}$$

## 3.3 User Activeness

The activeness of Twitter users is measured by the number of tweets they posted during a period. An active user is more likely to share valuable information, since he got more used to collecting interesting information, and sharing it. Figure 2 illustrates distribution of the number of users over the number of tweets they posted. It shows that the majority of users posted only one tweet, while the most active one posted 570 tweets in the corpus. With the number of posted tweets less than 12, the distribution follows the power law.

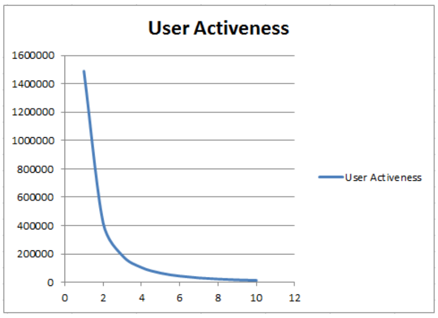

**Fig. 2.** user activeness

Let $u_i$ represent the number of posted tweets of user i. Thus the feature score $U_i$ for user i is calculated as follows.

$$U_i=\begin{cases}\ln u_i\ , & u_i\le 12\\ \ln 12, & u_i>12\end{cases}$$

## 3.4 Other Features

Tweet Length We use rule-based method known as WordNet is employed to deal with the stemming of nouns, verbs and adjectives. There are also lots of misspelled words in the tweets we also correct the words using method described in Wiktionary [20]. The length of a tweet is generally defined as the number of words after word stemming, skipping retweets, hashtags and urls. Figure 3 shows the tweets distribution over the length, which does not follow the power law. So we used the length directly as a feature.

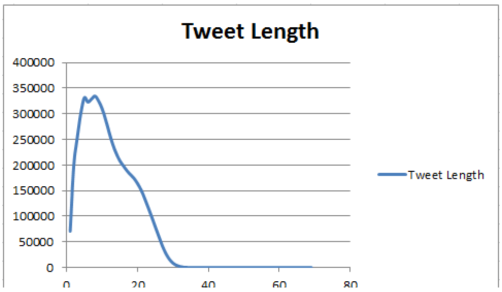

**Fig. 3.** Tweet length

The times of retweetness In Twitter, the times of retweetness reflect the popularity of a tweet. The relationship between the times of retweetness and the number of tweets is shown in Figure 4. We can see that a large number of tweets that are retweeted once, while only a few have been retweeted more than 20 times. The distribution for those retweeted follows the power law. Because of the power-law distribution, we use ln($r_i$+1) to denotes where $r_i$ is the times of retweetness of tweet i. $r_i$ equals zero if it is not retweeted.

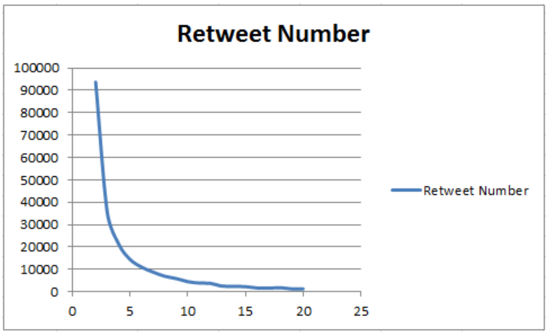

**Fig. 4.** Retweet Number distribution

URLs and Hashtags A tweet with urls usually contains more information than others. Thus the number of urls indicates the value of a tweet in our ranking model. Hashtags are designed to reflect the topic of a tweet, so we also consider them when determine the relevance. Since a hashtag maybe a combination of several words, we simply use substring match instead of whole word match as a query hits a hashtag. Similarly as the way we boost content match, the score is also boosted by multiplying the number of different query words got matched. The feature score $H_i$ of hashtags for tweet i is calculated as formula Where $q_j$ is the jth query word. |Q| is the total number of query words. And $h_j$,i is a 0-1 binary that indicates whether query word $q_j$ hits the hashtags in tweet i.

$$H_i = (\sum_{j=1}^{|Q|} IDF(q_j)\square h_{j,i})\square\sum_{j=1}^{|Q|} h_{j,i}$$

Freshness Because of we must keep the reverse time order, the freshness of a tweet is very important and may be more important than relevant. We use it as a feature. If the query time is T and the ith tweet's post time is $t_i$≤T, both are denoted by seconds from a given time point. We use ln(T-$t_i$+1) to measure the freshness of the tweet. The smaller the value is, the fresher the tweet.

## 4 Experiments and Evaluations

From the TREC Microblogging 2011's official result, we have got about labeled 38,482 tweets. We crawl twitter to get tweets they replied to and merge them up to get 48,543 tweets in total. Because we have the reply relationship among the tweets we can easily restore the conversational threads. We join up all the tweets in the same thread as a longer tweet to denote each single original tweet. So after that, we have 38,482 longer tweets. We compare the result of our method with simple method that determines the tweets relevance simply by whether they contain particular keywords or not. There are 50 topics in TREC's task and we use labeled tweets on 30 topics to train our model (topic-id from 1 to 30) and the other 20 topics (topic-id from 31 to 50) to test. For each test topic we give top30 results and compare. Table 3 shows the result. Using conversational threads can improve the precision@30 by about 29% in average. If we use tree thread height as a feature, the result will be even better. The second column show the result.

**Table 3.** Precision@30 compared with simple method and our method

| Topic-id | Our method Presion@30 | Using tree height a feature | Simple hit method |
|---|---|---|---|
| 31 | 2 | 4 | 1 |
| 32 | 1 | 2 | 1 |
| 33 | 8 | 8 | 0 |
| 34 | 1 | 2 | 0 |
| 35 | 1 | 1 | 3 |
| 36 | 13 | 13 | 9 |
| 37 | 0 | 0 | 1 |
| 38 | 2 | 2 | 1 |
| 39 | 8 | 8 | 7 |
| 40 | 1 | 1 | 2 |
| 41 | 1 | 1 | 1 |
| 42 | 1 | 1 | 2 |
| 43 | 4 | 5 | 2 |
| 44 | 3 | 6 | 0 |

| | | | |
|---|---|---|---|
| 45 | 2 | 2 | 2 |
| 46 | 1 | 1 | 9 |
| 47 | 0 | 0 | 0 |
| 48 | 2 | 2 | 0 |
| 49 | 2 | 3 | 0 |
| 50 | 0 | 0 | 0 |
| Total | 53 | 62 | 41 |

## 5 Conclusion and Future Work

Microblog tweets are always very short, which hinders big difficulty to searching for relevant tweets. To handle the problem, this paper proposed to employ conversational thread structure to expand short tweets and use an enhanced BM25 formula to determine the relevant and thus can find relevant more easily.

In this preliminary work, we simply join up all the tweets in conversational thread without consider to merge up hashtags and urls. The further study on the whole frame of combine tweet words, hashtags and urls together seems to be a good try.

## Acknowledgements

This work was supported by the National Natural Science Foundation of China with grant number 60933005 and the National Basic Research Program of China with grant number 2012CB316303. This work was also partly supported by the National Information Security Foundation of China with grant number 242-2011F45.

## References

1. http://en.wikipedia.org/wiki/Microblogging
2. http://www.howardyermish.com/2009/08/10/drinking-from-the-fire-hydrant/
3. A taxonomy of web search Andrei Broder IBM Research
4. Why We Twitter: Understanding Microblogging Usage and Communities Akshay Java, Xiaodan Song, Tim Finin, Belle Tseng Joint 9th WEBKDD and 1st SNA-KDD Workshop 2007
5. Analysis of Topological Characteristics of Huge Online Social Networking Services Yong-Yeo,l Ahn Seungyeop, Han Haewoon Kwak WWW2007
6. What is Twitter, a Social Network or a News Media? Haewoon Kwak, Changhyun Lee, Hosung Park, Sue Moon WWW2010
7. TwitterRank: Finding Topic-sensitive Influential Twitters Jianshu Weng, Ee-Peng Lim, Jing Jiang, Qi He WSDM2010
8. We Know Who You Followed Last Summer: Inferring Social Link Creation Times In Twitter Brendan Meeder, Brian Karrer, Amin Sayedi, R. Ravi Christian ,Borgs, Jennifer Chayes WWW2011

9. Differences in the Mechanics of Information Diffusion Across Topics: Idioms, Political Hashtags, and Complex Contagion on Twitter Daniel M. Romero, Brendan Meeder, Jon Kleinberg WWW2011
10. Who Says What to Whom on Twitter Shaomei Wu,Jake M. Hofman, Winter A. Maso, Duncan J. Watts WWW2011
11. Information Credibility on Twitter Carlos Castillo, Marcelo Mendoza ,Barbara Poblete WWW2011
12. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors Takeshi Sakaki,Makoto Okazaki,Yutaka Matsuo WWW2010
13. Using Twitter to Recommend Real-Time Topical News Kristina Lerman, Tad Hogg Rec-Sys 2009
14. Predicting the Future with Social Media[J]. Arxiv preprint arXiv:1003.5699. 2010. S. Asur, B. A. Huberman.
15. Finding Influetntials Based on the Temporal Order of Information Adoption in Twitter
Changhyun Lee, Haewoon Kwak, Hosung Park, Sue Moon
WWW2010
16. Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network Bongwon Suh, Lichan Hong, Peter Pirolli, Ed H. Chi ICSC2010
17. Online Community Search Using Thread Structure Jangwon Seo etc.al. CIKM2009
18. Search Using Social Media Structures Jangwon Seo etc.al. Phd Dissertation
19. Statistical Analysis of the Social Network and Discussion Threads in Slashdot Vicenç Gómez etc.al. WWW2008
20. http://en.wiktionary.org/wiki/Wiktionary:Frequency_lists