ICTNET at Microblog Track in TREC 2013

Jinhua Gao[1,2], Guoxin Cui[1,2], Shenghua Liu[1], Yue Liu[1], and Xueqi Cheng[1]

[1]Center of Web Data Science and Engineering,
Institute of Computing Technology, Chinese Academy of Sciences,
Beijing, China, 100190

[2]University of Chinese Academy of Sciences,
Beijing, China, 100190

1 Introduction

Microblogging services, in which users can publish and share personal status, are now very popular, and attracting more and more industrial and scientific interests. Twitter is one of the most famous microblogging services. On twitter, Users can post personal updates, which are called tweets and limited to 140 characters long. In tweets, users can share interesting messages to their friends by retweeting(RT), push some tweets to specific users using @ mention, and indicate the topics of their tweets using # hashtags. The short-text characteristic and social attributes such as RT, @ mention and # hashtags, make traditional problems, like search, rank, and recommendation etc, quite different in microblogging settings.

Microblog track was first introduced in 2011, and ICTNET group have participated in this track three times[2, 7]. In this year's track, only the realtime ad-hoc search task, which was also proposed in 2011 and 2012, was open for submission. This task requires to retrieve all the tweets that are relevant to query **Q** and before time **T**. Unlike the past two years, in which participants had to collect the corpus themselves, microblog track was first provided as a service this year, and participants could access the corpus through official APIs, which made it possible for the organizers to increase the size of corpus from 16M tweets to 260M tweets, which were collected via the Twitter streaming API over a two-month period.

This report is organized as follows. Section 2 mainly focuses on the data preparation step, which contains the data collecting step and preprocessing step. The methodology and framework are illustrated in section 3, and some experiments and results are presented in section 4.

2 Data Preparation

We downloaded the twitter-tools[5] from github to interact with the service API. The service API simply returned relevant tweets' information, which included tweets text and some of their social attributes such as followers count, whether or not being a reply, and retweeted count, for submitted queries. As the number of submitted tweets for each query was limited to 1000, we retrieved 10,000 tweets for each query, and stored their details in files for later analysis. We also stored tweets for topics of 2011 and 2012 microblog tracks, since they were used as training data for our supervised framework.

Word stemming was processed using the *TweetAnalyzer* class extracted from the twitter-tools. Each tweet was split into terms, and terms whose length were less than 3 were dropped. Non-English tweets were also filtered. Stop words removal was not processed, as it has been shown that stop words removal might have a negative impact on the ranking results in short-text settings.

3 Methodology

We defined this task to be a re-rank problem. We could obtain the initial ranking list through the service API, and re-rank the list to achieve our final ranking. However, the short-text characteristic and social attributes of tweets made traditional ranking method not work well. Firstly, the tweets and queries were both so short that traditional ranking model could not distinguish relevant tweets from non-relevant tweets well, which could resulted in a poor initial list, and thus influence the final ranking. Secondly, it worth studying how to incorporate the social attributes into ranking.

In order to obtain a better initial list, we expanded the query using pseudo relevance feedback, and submitted the expanded query to the service API. The returned list were applied to re-ranking. To address the second problem, SVM Rank model[4] was applied, in which those social attributes were transformed into training features.

3.1 Query Expansion

Pseudo relevance feedback is a traditional method to expand the queries. It assumes the top k documents in the returned list to be relevant, and extracts the terms that have higher weight under some weight function settings to be the expanded terms. In our method, we took top 30 tweets for expansion, and 5 terms were expanded for each query. We first checked some traditional weight functions, including *tf*, *idf*, and *tf * idf*. We manually checked the expanded terms, and those results turned out to be rather poor.

We further investigated the Divergence From Randomness(DFR) term weighting model[3], whose main idea is to weight the informativeness of a term by the divergence of its distribution in the top-ranked documents from a random distribution. We chose the Bo1 model[1], which utilized the Bose-Einstein statistics. In this model, the weight w of a term t is given by:

$$w(t) = tf \cdot log_2 \frac{1 + P_n}{P_n} + log_2(1 + P_n) \tag{1}$$

where tf is the term frequency in top-ranked documents, and P_n is given by $\frac{F}{N}$ with F representing the term frequency in the whole collection and N indicating the number of documents in the collection.

Besides, we tried another term weighting model which took word co-occurrence into account. The relevance score of a term in a tweet was inferred by a function of the number of query term hits in that tweet, and the weight of a term was gained by summing its scores over all the top-ranked tweets. To choose the score function, we heuristically tried the linear function, in which the score was exactly the number of query term hits, and the exponential function, where the score was equal to exponential result of the number of query term hits, since we thought the more query terms it co-occurred with in a tweet, the more relevant it would be. The latter one gave better results, and was adopted in our experiment. This model was referred as the co-occurrence model.

3.2 Feature Extraction

To train the SVM Rank model, a set of features was extracted for each tweet, which could be split into two views: relevance view and intrinsic view. Features of relevance view were exactly the same as those in traditional documents ranking, as were reported in LETOR[6], which was a benchmark dataset released by Microsoft Research Asia for learning to rank research. Those features were query-dependent, including term frequency(*tf*), inverse document frequency(*idf*), BM25 score, and language model score etc.

The features of intrinsic view were query-independent, and those social attributes of tweets such as @ mentions, # hashtags, and retweeted count were incorporated. The whole features extracted were listed in table 1. They were quite self-explained by their names, in which names ending with "_exd" meant features calculated using expanded terms only.

There were totally 44 features to be extracted for each tweet, and those features were all query-based normalized for SVM rank.

Table 1: Features extracted for ranking

Intrinsic View	Relevance View
tweet_length	tf_sum, tf_min, tf_max, tf_avg
hasRT	tf_sum_exd, tf_min_exd, tf_max_exd, tf_avg_exd
number_of_hashtags	idf_sum, idf_min, idf_max, idf_avg
number_of_at_mentions	idf_sum_exd, idf_min_exd, idf_max_exd, idf_avg_exd
number_of_urls	normalized_tf_sum
retweeted_count	tf_idf_sum, tf_idf_min, tf_idf_max, tf_idf_avg
follower_count	tf_idf_sum_exd, tf_idf_min_exd, tf_idf_max_exd, tf_idf_avg_exd
number_of_status	bm25_score, VSM_score, covered_term_ratio
unique_term_ratio	bm25_score_exd, VSM_score_exd, covered_term_ratio_exd
	retrieval_score

Table 2: Query expansion results

Queries	Expansion Method	
	Bo1 Model	Co-occurrence Model
2022 fifa soccer	cup blatter qatar winter presid	plai cup 360 game hour
oprah winfrei half sister	secret famili reveal exist promi	secret famili love watch time
carbon monoxid law	poison detector famili car warn	health watch firm school judge
bedbug epidem	warn femal ruin specialist war	obes bite warn kid spread
michel obama obes campaign	ladi move atlanta childhood press	bachman ladi love design union
iran nuclear program	talk afp close 2012 power	egypt talk power obama iranian

3.3 Applying SVM Rank

SVM Rank model was adopted to obtain the final ranking. Official released relevance judgements for microblog track 2011 and 2012 were utilized as labelled training data, which consisted of 108 queries with tweets labelled using 0, 1, 2, meaning non-relevant, relevant and highly-relevant respectively. During model training process, linear kernel was adopted, and 5-fold cross validation was applied to tune the parameter C, which controlled the balance between training error and margin.

4 Experiment

We tried out two query expansion methods, and the results were shown in table 2.

It's easily seen that terms expanded using those two models shared some similarities, but the performance varied among different queries. The query "2022 fifa soccer" mainly talked about Qatar winning the bid to host 2022 world cup. Bo1 model worked fine, but co-occurrence model transferred to "fifa soccer game" topic. And the result was totally opposite on the case "bedbug epidem", which was about epidemic caused by some bedbugs. ICTNETRUN2 was based on co-occurrence model, while ICTNETRUN3 was obtained using bo1 model.

To tune the parameters C of SVM Rank model, we conducted 5-fold cross validation on 2011 and 2012's queries respectively. The result was shown in figure 1. We finally set C to be 0.30. Official relevance judgements of microblog track 2011 and 2012 were combined to serve as labelled data, and SVM Rank model was trained and applied to this year's queries to obtain our submitted runs.

Table 3: Evaluation results for ICTNET submitted runs

Runtag	Precision @ 30	
	relevant	highly relevant
ICTNETRUN1	0.3378	0.1683
ICTNETRUN2	0.4594	0.2289
ICTNETRUN3	0.4644	0.2367

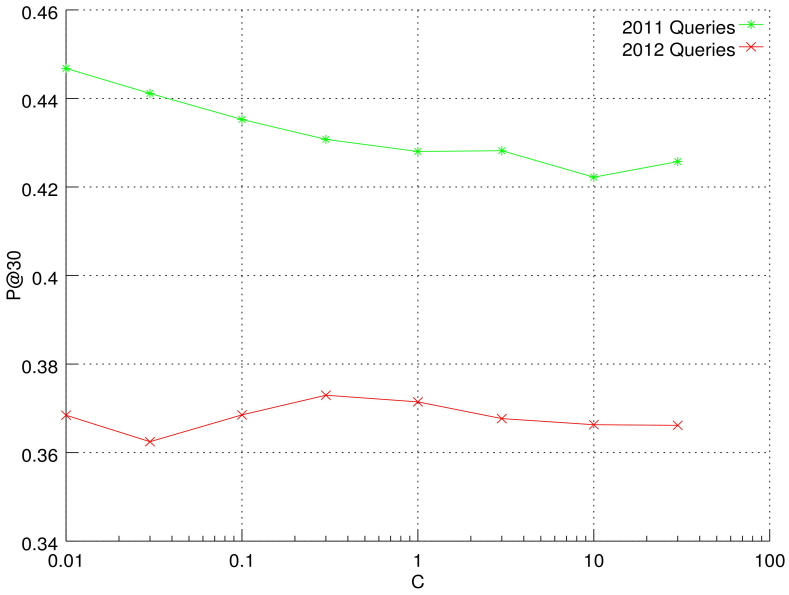

Figure 1: Parameter selection for SVM rank

The relevance judgement for microblog track 2013 has been released when this report was written, and the results of our runs were demonstrated in table 3.

Acknowledgements

We would like to thank all the organizers and assessors of TREC and NIST. This work is sponsored by NSF of China Grants No.61100083, 242 Program of China Grants No.2013F97, and the National Key Technology R&D Program(2012BAH39B04).

References

[1] Giambattista Amati. *Probabilistic Models for Information Retrieval Based on Divergence from Randomness*. PhD thesis, Department of Computing Science, University of Glasgow, 2003.

[2] Peng Cao, Jinhua Gao, Yubao Yu, Shenghua Liu, Yue Liu, and Xueqi Cheng. ICTNET at Microblog Track TREC 2011. In *TREC*, 2011.

[3] Ben He and Iadh Ounis. Combining Fields for Query Expansion and Adaptive Query Expansion. *Information Processing and Management*, 43:1294–1307, 2007.

[4] Thorsten Joachims. Training Linear SVMs in Linear Time. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

[5] Jimmy Lin. Twitter Tools. `https://github.com/lintool/twitter-tools`, 2012.

[6] Tieyan Liu, Jun Xu, Tao Qin, Wenying Xiong, and Hang Li. LETOR: Benchmark Dataset for Research on Learning to Rank for Information Retrieval. In *Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, 2007.

[7] Bolong Zhu, Jinhua Gao, Xiao Han, Cunhui Shi, Shenghua Liu, Yue Liu, and Xueqi Cheng. ICTNET at Microblog Track TREC 2012. In *TREC*, 2012.