# Feature Analysis in Microblog Retrieval Based on Learning to Rank

Zhongyuan Han[1,2], Xuwei Li[1,*], Muyun Yang[1], Haoliang Qi[2], and Sheng Li[1]

[1] School of Computer Science and Technology,
Harbin Institute of Technology, Harbin, China
{zyhan,xwli,ymy}@mtlab.hit.edu.cn, lisheng@hit.edu.cn
[2] School of Computer Science and Technology,
Heilongjiang Institute of Technology, Harbin, China
haoliang.qi@gmail.com

**Abstract.** Learning to rank, which can fuse various of features, performs well in microblog retrieval. However, it is still unclear how the features function in microblog ranking. To address this issue, this paper examines the contribution of each single feature together with the contribution of the feature combinations via the ranking SVM for microblog retrieval modeling. The experimental results on the TREC microblog collection show that textual features, i.e. content relevance between a query and a microblog, contribute most to the retrieval performance. And the combination of certain non-textual features and textual features can further enhance the retrieval performance, though non-textual features alone produce rather weak results.

**Keywords:** microblog retrieval, learning to rank, feature combination.

## 1 Introduction

Current retrieval models are usually built on so-called learning-to-rank strategy, which typically involves multiple features from the queries and the documents. This strategy has also been applied to microblog retrieval [1]. In TREC microblog track, both the USC/ISI team (top 1 in TREC 2011) and the HIT team (top 1 in TREC 2012) used Learning to Rank algorithm [2,3]. Several other teams also adopted similar methods [4-8], differing only in the different features employed.

In the literature, however, the features for microblog ranking have not been well examined. Besides the classical textual features, microblog retrieval is further enriched by various non-text features, which has been proved to be more effective. Duan et al employed learning to rank algorithms to determine the best set of features, in which the textual features hardly contribute to the retrieval performance [1].

Following this thread, the feature contribution is reexamined in microblog retrieval in this paper, including the single feature and the feature combination via the Ranking SVM framework. Specifically, we focus on the textual features, i.e. the content relevance between the query and the microblog.

* Corresponding author.

G. Zhou et al. (Eds.): NLPCC 2013, CCIS 400, pp. 410–416, 2013.

The rest of this paper is organized as follows: First, the features in Ranking SVM for microblog retrieval are introduced. Second, the experiment and evaluation are given. Last, we draw a conclusion for this paper and future direction in this field is discussed.

## 2 Features in Ranking SVM for Microblog Retrieval

Ranking SVM, one of the pair-wise ranking methods, is an application of support vector machine, which is used to solve certain ranking problems. In microblog retrieval, the training data is a set of $(x_{u,v},y_{u,v})$. $x_{u,v}$ is a microblog pair(u,v). Here u and v indicate a microblog presented by a feature vector. If u<v, $y_{u,v}$=1 ; otherwise $y_{u,v}$=-1. It means that the train sample is positive if the microblog u has a higher relevant level than microblog v. Thus the ranking task is changed into a classification task. The retrieval model can be trained by SVM.

In learning to rank, the feature set is crucial to the model performance. To determine the contribution of each feature, a common practice is to re-build the model by each single feature as well as different feature combinations in addition to the whole feature set. The differences in the model performances are then deemed as a good proof for the feature influence in the retrieval modeling.

In this paper, we classify the features for microblog ranking into three groups: content relevance features, author features and microblog unique features, which are detailed in the following section.

### 2.1 Content Relevance Features

Content relevance features, often referred as the textual features, specify the content relevance between queries and tweets. Under language model framework, here we use Kullback-Leibler Divergence to measure the content relevance between query model *Q* and microblog model *M*. The standard KL function is:

$$KL(Q \mid M) = \sum_{w} P(w \mid M) \log \frac{P(w \mid M)}{P(w \mid Q)} \tag{1}$$

Then four content relevance features can be obtained as shown in Table 1.

**Table 1.** Content relevance features

| Features | Description |
|---|---|
| KL_OQ_OM | KL score of original query and original microblog |
| KL_EQ_OM | KL score of expanded query and original microblog |
| KL_OQ_EM | KL score of original query and expanded microblog |
| KL_EQ_QM | KL score of expanded query and expanded microblog |

OQ is denoted as original query model and OM is denoted as original microblog model. For short queries and short microblogs in microblog retrieval, the query expansion and microblog expansion are used to estimate query model (denoted by EQ) and microblog model (denoted by EM).

**OQ (Original Query) and OM (Original Microblog)** are estimated by maximum likelihood estimation on original query and original microblog.

**EQ (Expanded Query)** is modeled by the relevance feedback model [9]. According to the relevance model, a query term is generated by a relevance model $p(w|\theta_R)$, which is derived by top-ranked feedback documents by assuming them to be samples from the relevance model as follows:

$$p(w \mid \theta_R) \propto \sum_{d \in F} p(w \mid d)\, p(d \mid \theta_R) \tag{2}$$

where $F$ denotes the feedback documents, usually approximated by the top-ranked retrieved documents for the query; $p(w|d)$ is the probability that the term w appears in the document d, and $p(d|\theta_R)$ is the probability that d is generated by $\theta_R$. $\theta_R$ is estimated by the original query, thus we can obtain:

$$p(w \mid \theta_R) \propto \sum_{d \in F} p(w \mid \theta_R)\, p(\theta_R) \prod_{i=1}^{m} p(q_i \mid \theta_R) \tag{3}$$

The above relevance model is used to enhance the original query model by the following interpolation:

$$p(w \mid \theta_q') = (1-\alpha)\, p(w \mid \theta_q) + \alpha\, p(w \mid \theta_R) \tag{4}$$

where $\alpha$ is the interpolation weight. In our experiments, $\alpha$=0.8 and the number of top-ranked retrieved documents is set 20.

**EM (Expanded Microblog)** is estimated by DELM (Document Expansion Language Model) [10] to improve the representation of short tweets. That is, for a document d (i.e. tweet), decide its k (set as 100 in our experiment) nearest neighbors $\{b_1,\ldots,b_k\}$ by the cosine similarity score between $b_k$ and $d$. Then it assigns a confidence value $r_d(b)$ to every document $b$ to indicate our confidence about that $b$ is sampled from $d$'s hidden model. The confidence value is defined as below:

$$r_d(b) = \frac{sim(d,b)}{\sum_{b' \in C-\{d\}} sim(d,b')} \tag{5}$$

In fact, the confidence value $r_d(b)$ is set by normalizing the cosine similarity scores. Then a pseudo document $d'$ is obtained with the following pseudo term count:

$$c(w,d') = \beta c(w,d) + (1-\beta) \sum_{b \in C-\{d\}} (\gamma_d(b) \times c(w,b)) \tag{6}$$

where parameter $\beta$ (set as 0.8 in our experiment) controls the degree of relying on neighborhood document. This technique is proved to be valid in improving search results in TREC texts by [6].

### 2.2 Author Features and Microblog Unique Features

Author features, listed in table 2, reflect the publisher of a tweet.

**Table 2.** Author features

| Features | Description |
|---|---|
| FOLLOWERS_COUNT | How many people are following this author |
| FRIENDS_COUNT | How many people this author is following |
| LISTS_COUNT | How many groups is the author in |
| STATUS_COUNT | How many microblogs are posted by the author |
| FAVOURITE_COUNT | How many microblogs are the author's favorite |
| IS_VERIFIED | Is the author verified |

Microblog unique features refer to the particular characteristics of a tweet, which are summarized in Table 3.

**Table 3.** Microblog unique features

| Features | Description |
|---|---|
| HAS_URL | Whether the microblog contains a URL |
| IS_REPLY | Whether the microblog is a reply microblog |
| HAS_MENTION | Whether the microblog contains a mention("@") |
| HAS_HASHTAG | Whether the microblog contains a hashtag("#") |
| RETWEET_COUNT | How many is retweet count |

### 2.3 Feature Sets

Several feature sets mentioned above are examined in the subsequent experiments. The performance of the model built by all features (RankSVM_all) is denoted as the baseline, and the varied feature settings consist of the following:

**Leave_one_out_from_all**, in which each single feature is removed respectively from the total 15 features, demonstrates the contribution of each feature in the feature set.

**Single_feature**, in which only one feature is involved to model the microblog retrieval**,** is used to reveal the importance of each feature in the model alone.

**Feature_group** is used to examine different kinds of features in three groups. This setting compares the different aspects of features existing in microblogs to some extent.

**Best_feature** is an optimized subset with the best retrieval performance. We generate several feature sets randomly and use the advanced greedy feature selection method proposed by Duan et al [1], to find the best feature combinational set with the best performance.

# 3 Experiment and Evaluation

## 3.1 Experimental Settings

The experiment data is TREC 2011 tweets corpus. The corpus, which is comprised of 2 weeks tweets sampled from Twitter, contains about ten million tweets[1]. We download 10,397,336 tweets by twitter crawler provided by track organizers, and there remain 3,754,077 tweets for the experiment after being filtered in accordance with TREC [11]. The statistics of dataset are shown in Table 4. Note that the index is built for each query with only tweets before its query time.

**Table 4.** Statistics of tweets in dataset

| # of Total tweets | # of Null tweets | # of Retweets | # of Non-English tweets | # of Indexed Tweets |
|---|---|---|---|---|
| 10,397,336 | 0 | 342,652 | 6,300,607 | 3,754,077 |

The 50 queries and the corresponding answer sets in TREC 2011 are used to train the retrieval model via SVMrank[2] by Thorsten Joachims. The 60 queries of TREC 2012 are the test set. Following TREC 2012 microblog tack, the performance is evaluated by standard metrics: P@30. Meanwhile, MAP and R-Prec are reported for reference.

## 3.2 All Features vs. Single

As is shown in table 5, the performance of RankSVM_all is better than that of any single feature. Although any feature from the author aspect and the tweet unique produce poor result, they can enhance the text relevance features as a whole to achieve a significantly better result.

We further examine the feature contribution by removing one from the all-set respectively, and the corresponding results are shown in Fig. 1. According to this figure, most features cause a performance drop if they are removed. The most significant drops occur with the removal of the HAS_URL feature and the KL_EQ_EM feature. The HAS_MENTION is harmful for ranking.

We then examine the features in three groups and compare them with the best feature set achieved. According to Table 5, it is revealed that the performance of each feature group is inferior to that of the all features. The best feature set can boost the retrieval performance from 0.2593 to 0.2621($p<0.05$) in P@30, with the following 9 features left in the core: 4 content relevance scores, FOLLOWERS_COUNT, LISTS_COUNT, HAS_URL, HAS_HASHTAG and RETWEET_COUNT.

A notable finding in the experiment is that the content relevance features are strong indicators for tweet retrieval performance. This fact indicates that the content relevance between query and tweet is still essential to tweet retrieval performance. In addition, the expansion techniques consistently exhibit positive effectiveness in performance improvement.

[1] https://ir.nist.gov/tweets2011/id-status.01-May-2012.gz
[2] http://www.cs.cornell.edu/people/tj/svm_light/svm_rank.html

**Table 5.** Performance comparisons of using all features vs. single feature

| | P@30 | MAP | R-Prec |
|---|---|---|---|
| RankSVM_all | 0.2593 | 0.2475 | 0.2684 |
| RankSVM_best | **0.2621** | **0.2479** | **0.2685** |
| KL_OQ_OM | 0.2062 | 0.1855 | 0.2207 |
| KL_EQ_OM | 0.2345 | 0.2302 | 0.2471 |
| KL_OQ_EM | 0.226 | 0.2059 | 0.2345 |
| KL_EQ_EM | 0.2446 | 0.2356 | 0.2575 |
| Group_content_relevance | 0.2458 | 0.2371 | 0.2569 |
| FAVOURITE_COUNT | 0.0181 | 0.0225 | 0.0204 |
| FRIENDS_COUNT | 0.0418 | 0.0339 | 0.0407 |
| FOLLOWERS_COUNT | 0.0542 | 0.0511 | 0.0555 |
| LISTS_COUNT | 0.0435 | 0.0361 | 0.0392 |
| STATUS_COUNT | 0.0475 | 0.0533 | 0.0494 |
| IS_VERIFIED | 0.0542 | 0.042 | 0.0533 |
| Group_Author | 0.048 | 0.0413 | 0.0452 |
| RETWEET_COUNT | 0.0429 | 0.0356 | 0.0378 |
| IS_REPLY | 0.0469 | 0.0531 | 0.0494 |
| HAS_MENTION | 0.0486 | 0.0554 | 0.0524 |
| HAS_URL | 0.0706 | 0.0715 | 0.0758 |
| HAS_HASHTAG | 0.0345 | 0.0363 | 0.0399 |
| Group_Unique | 0.0695 | 0.0554 | 0.065 |

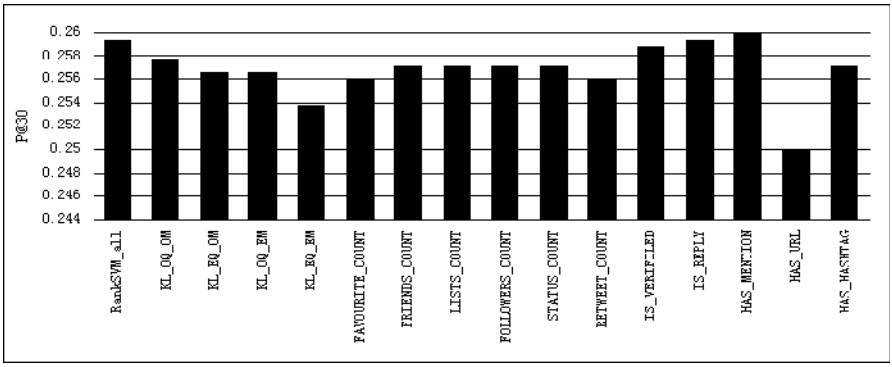


**Fig. 1.** Performance of leave_one_out_from_all

## 4 Conclusion and Future Work

This paper investigates the feature contribution in microblog retrieval modeling under learning-to-rank framework. 15 features, which can be classified into three groups: text relevance, author and tweet unique, are examined via Ranking SVM. Experiment results show that the most important features are content relevance features, which lie in the core for the model performance. Both query expansion and microblog

expansion are the most important features in content features. Meanwhile, the non-textual features could enrich the text relevance scores, though they produce unsatisfactory results alone. Among non-textual features, the HAS_URL produces the most significant effect to the performance.

In future work, we would explore how to combine the content relevance scores with the non-textual features. In addition, certain factors which have not covered in this paper, such as time influence to microblog search, should also be addressed.

**Acknowledgements.** This work is supported by the NSF China (No. 61272384 & 61105072), and the National High Technology Research and Development Program of China (863 Program, No. 2011AA01A207).

## References

1. Duan, Y., Jiang, L., Qin, T., Zhou, M., Shum, H.Y.: An Empirical Study on Learning to Rank of Tweets. In: Proceedings of the 23rd International Conference on Computational Linguistics, pp. 295–303. Association for Computational Linguistics, Beijing (2010)
2. Metzler, D., Cai, C.: USC/ISI at TREC 2011: Microblog Track. In: Proceeding of the Twentieth Text REtrieval Conference. NIST, Gaithersburg (2011)
3. Han, Z., Li, X., Yang, M., Qi, H., Li, S., Zhao, T.: HIT at TREC 2012 Microblog Track. In: Proceeding of the Twenty-First Text REtrieval Conference. NIST, Gaithersburg (2012)
4. Zhang, X., Lu, S., He, B., Xu, J., Luo, T.: UCAS at TREC-2012 Microblog Track. In: Proceeding of the Twenty-First Text REtrieval Conference. NIST, Gaithersburg (2012)
5. Zhu, B., Gao, J., Han, X., Shi, C., Liu, S., Liu, Y., Cheng, X.: ICTNET at Microblog Track TREC 2012. In: Proceeding of the Twenty-First Text REtrieval Conference. NIST, Gaithersburg (2012)
6. Liang, F., Qiang, R., Hong, Y., Fei, Y., Yang, J.: PKUICST at TREC 2012 Microblog Track. In: Proceeding of the Twenty-First Text REtrieval Conference. NIST, Gaithersburg (2012)
7. Berendsen, R., Meij, E., Odijk, D., de Rijke, M., Weerkamp, W.: The University of Amsterdam at TREC 2012. In: Proceeding of the Twenty-First Text REtrieval Conference. NIST, Gaithersburg (2012)
8. Miyanishi, T., Okamura, N., Liu, X., Seki, K., Uehara, K.: Trec 2011 Microblog Track Experiments at Kobe University. In: Proceeding of the Twentieth Text REtrieval Conference. NIST, Gaithersburg (2011)
9. Lavrenko, V., Croft, W.B.: Relevance-Based Language Models. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 120–127. ACM, New York (2001)
10. Tao, T., Wang, X., Mei, Q., Zhai, C.: Language Model Information Retrieval with Document Expansion. In: Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 407–414. Association for Computational Linguistics, New York (2006)
11. Ounis, I., Macdonald, C., Lin, J., Soboro, I.: Overview of the TREC-2011 Microblog Track. In: Proceeding of the Twentieth Text REtrieval Conference. NIST, Gaithersburg (2011)