International Journal of Grid and Distributed Computing
Vol. 9, No. 1, 2016, pp.89-98
http://dx.doi.org/10.14257/ijgdc.2016.9.1.09

A Temporal Microblog Filtering Model

Zhongyuan Han[1, 2], Muyun Yang[1], Leilei Kong[2], Haoliang Qi[2] and Sheng Li[1]

*1 School of Computer Science and Technology,
Harbin Institute of Technology, Harbin, China;
2 School of Computer Science and Technology,
Heilongjiang Institute of Technology, Harbin, China;
hanzhongyuan@gmail.com, ymy@mtlab.hit.edu.cn, kongleilei1979@hotmail.com,
haoliang.qi@gmail.com, lisheng@hit.edu.cn*

Abstract

The rapid growth in the popularity of social networking and microblogging has led to a new way of finding and broadcasting information in the recent years. The real-time microblog filtering emerges as the times require. The task of real-time microblog filtering is to decide if subsequently posted tweets are relevant to a given query which represents the special information needs. One-side feedback is one of the most difficult problems in microblog filtering. This paper focuses on exploiting the time profile of relevant microblogs to address this problem. A temporal microblog filtering based on retrieval model is proposed. Specifically, similarity threshold achieved by the language model is adjusted according to temporal burst. Evaluated on the TREC 2012 microblog real-time filtering track dataset, the experimental results show that the proposed model is significantly better than several baselines.

***Keywords**: real-time filtering; filtering model; information retrieval; microblog*

1. Introduction

The boom of various social media has facilitated information creation, sharing, and diffusion among users. Microblog retrieval has become important information resource for many online users. In the study of microblog field, microblog filtering is a hot research topic in recent years. It can be used in real-time monitoring[1], topic tracking[2], microblog recommending[3], improving tweet search performance[4], hiding content about a very specific given topic[5], and so on.

To explore the search behavior and boost the search performance in the real-time environment, TREC (Text RetriEval Conference) introduced a novel pilot track named microblog track. As an authoritative evaluation conference in the field of text retrieval, TREC describes the task of the microblog real-time filtering: a query arrives at a defined point in time, and the system must filter the subsequent stream of messages to select tweets relevant to the information need[6].

A typical scenario of microblog filtering is: the filter processes the tweets from the query-tweet-time to the query-newest-time, one at a time and shows the tweet which is decided to be relevant to the query to the user. If the system decides to show the tweet, it could access the tweet's relevance judgment (if any) as immediate relevance feedback, but not otherwise [6]. In other words, the filter could receive the feedback from the user about whether a delivered microblog is relevant to the query during the filtering.

The scenario of microblog filtering shows that the feedback in the microblog real-time filtering is one-side feedback [7]. It means only the microblogs, which are judged to be relevant by filtering system, can get the feedback information. For the task of microblog filtering, the one-side feedback is one of the most intractable problems. It can cripple the performance of classical online learners.

ISSN: 2005-4262 IJGDC

The filtering task is similar to the ad-hoc retrieval task[8]. It is exactly the reverse of the real-time search task. Algorithms based on retrieval model, such as LM (language model), BM25, VSM(Vector Space Model), Rocchio, etc., can be typically used in this task[9−13]. Previous studies show that some filtering algorithms based on retrieval model work well for the sake of very few training data available at the early stage[8] and one-side feedback. Since the retrieval models produce the better performance, we follow the studies of retrieval model-based filter in this paper.

In recent years, exploiting the temporal to boost the performance of microblog retrieval has become one of the hot issues in the field of microblog search[14-18]. A typical practice in existing studies is to re-estimate the term weight or document prior probability by introducing a temporal prior. However, the time profile has not been applied in the studies of microblog filtering based on retrieval model directly. The main reason is the difficulties of estimating the time prior. In the environment of microblog filtering, we cannot get the whole distribution information of relevant documents to estimate the prior on the whole microblog collection which is unseen to the filter.

D.Sculley has shown that once the hyperplane has been shifted towards the positive side, it can never be shifted back[7]. Some examples located at the negative side are chosen to learn/be learned?, so the hyperplane can move towards the negative side[7]. To address the one-side feedback, in microblog filtering, considering the effectiveness of time profile, we propose a novel temporal microblog filtering model based on retrieval model. Specifically, the time profile is integrated into the retrieval model based on language model to help the filter to move the hyperplane toward the negative side. Several retrieval techniques, such as smoothing, pseudo feedback, etc., are employed to initialize and optimize the filter.

The temporal microblog filtering model has been experimented on TREC 2012 microblog real-time filtering track. The experimental results show that the proposed model is significantly better than both the microblog filters and get the best methods in the TREC 2012 microblog real-time filtering track.

The rest of this paper is structured as follows. In section 2, we provide a description of the real-time tweet filtering task and present the related work. Section 3 introduces the Temporal Microblog Filtering Model. Experimental results are presented in section 4. Finally, the paper is discussed and concluded in section 5.

2. Related Work

2.1. Real-Time Microblog Filtering Task

The popularity of microblog has increased information searching behaviors in the microblog environments. To boost the search performance in the real-time environment, TREC introduced a novel track named microblog track in 2012.

The real-time filtering task aims at deciding if subsequently posted tweets are relevant to a query which arrives at a particular time point. In this task, the user is interested in new relevant tweets, thus to keep up to date about a developing topic. For a specific microblog, if the filtering system makes a positive decision, the system will be allowed to know whether the microblog is truly relevant. If the system emits a negative decision for a microblog, it will not get any feedback information [6] .

2.2. Microblog Filtering Based on Retrieval Model

In the real-time microblog filtering task, the filtering algorithms based on retrieval model(*e.g.* language model) are often used. The algorithms based on retrieval model rank the microblogs and make decisions by comparing the similarity score with a threshold. Figure 1 shows the basic ideas of these approaches:

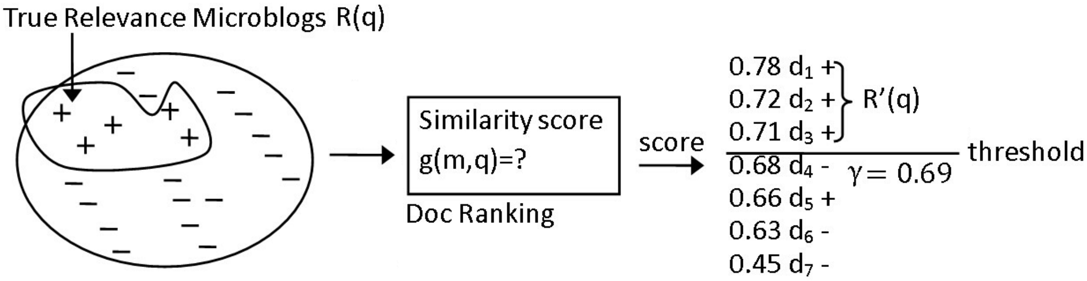

Figure 1. Filters Based on Retrieval Model

The core principle of these algorithms is how to update the query and the threshold according to the feedback. Karimi *et al.* use pseudo-relevance feedback to expand the queries [13]. Albakour *et al.* focus on query drift by which the users interests were modified to balance the importance of the short term interests, such as emerging subtopics, and the long-term interests in the overall topic [19]. Limsopatham *et al.* use the extended query to enrich the center of Rocchio feedback algorithm [10] . Other studies focus on how to update the threshold. A fixed threshold is used in Ref.[20]. Din *et al.* aim at optimizing $F_{0.5}$ to update the threshold [21]. And the k-th microblog similarity score in ranking list is set as the threshold [11, 13]. These algorithms lack the effective strategies for reducing the threshold to address the one-side feedback. So this paper explores how to update the threshold of the retrieval model-based filter for microblog real-time filtering task.

3. Temporal Microblog Filtering Model

In this section, we introduce the temporal microblog filtering model. Temporal microblog filtering model focuses the one-side feedback in microblog filtering and integrates the time profile into filter. In our temporal microblog filtering model, the basic filtering phases can be described as follows. Firstly, we construct the microblog model θ_m and query model θ_q, and then compute the similarity between θ_m and θ_q. The relevance of microblog and query would be decided by comparing the similarity score and the threshold. Then, the query model will be re-estimate according to the results of user feedback and the weight will be updated. Lastly, the most important part of our filtering framework is the threshold of regeneration. A novel time-based threshold of regeneration approach is proposed to combine time profile into the filtering framework.

3.1. Temporal Microblog Filtering Model Framework

In our temporal microblog filtering model framework, the relevance between the microblog and the query is decided as follows:

$$\mathrm{h}(\theta_q,\theta_m,\gamma,t)=\begin{cases}1 & \text{if } \mathrm{sim}(\theta_q,\theta_m)>\gamma(t)\\ 0 & \text{else}\end{cases} \tag{1}$$

where, θ_q is the query model, θ_m is the microblog model which need to be judged. $\mathrm{sim}(\theta_q,\theta_m)$ is the similarity score between θ_q and θ_m. $\gamma(t)$ is a threshold and is used to compare with $\mathrm{sim}(\theta_q,\theta_m)$. It will be re-estimate by using time profile which is described in detail in Section 3.5. If $\mathrm{sim}(\theta_q,\theta_m)$ is greater than $\gamma(t)$, then the microblog m will be regarded as a relevant microblog, but not otherwise.

3.2. Estimating γ(t)

As previously mentioned and shown in Figure 1, a microblog rank can be gotten by retrieval model and the threshold γ is used to separate the relevant microblogs from irrelevant microblogs. We assume that the similarity scores of the relevant microblogs are greater than that of the irrelevant ones. Therefore the score of the k-th microblog is an ideal γ if k is set as the number of relevant microblogs(NRM), shown in formula 11.

$$\gamma(t) = find\_kth(NRM(t)) \qquad (2)$$

Where NRM (t) is the number of relevant microblogs at the time t and find_kth(·) is a function returning the k-th max similarity score.

How to forecast NRM is the key issue of the γ selection under the above assumption. If the feedback information for all microblogs is available, NRM equals to NRM + 1 when a new relevant microblog appears. But in the real-time microblog filtering task, one-side feedback makes that only the samples judged positive by the filter could get the feedback information. It means that we will not know the number of the lost relevant microblogs once they are misjudged as irrelevant. Therefore, how to predict NRM is the key of one-side feedback problem in Temporal Microblog Filtering Model.

Previous works have shown that the most relevant microblogs are usually posted in a short period and the performance of the microblog retrieval model can be improved by using time distribution[14−18]. These time-based retrieval models are not used in microblog filtering task directly because we cannot evaluate the time distribution of relevant microblogs owing to the fact that the microblogs are posted one by one. Therefore, we predict NRM at microcosmic level. At microcosmic level, many relevant microblogs in some periods of time means that the probability of there being many relevant microblogs near a relevant one is higher than that of there being relevant microblogs near an irrelevant one. When a relevant microblog m is posted on time t_m, we suppose the count of relevant microblogs(NRM) is influenced by microblog m. Considering the previous work, we suppose the influence of one relevant microblog is a Gaussian function[14].

$$\Delta NRM(t,t_m) = \lambda e^{\frac{-(t-t_m)^2}{2\varphi^2}} + 1 \qquad (3)$$

where t_m is the post time of the relevant microblog m (in terms of seconds). t is the current time (in terms of seconds). λ and φ are the parameters need to be estimated. φ controls the influence extent, and λ determines the degree of influence. Because microblog m itself is a relevant microblog, ΔNRM must be larger than 1. ΔNRM will grow smaller as time goes by.

At the time point t, NRM is an accumulation of ΔNRM of all known relevant microblogs. So we have:

$$NRM(t) = NRM_0 + \sum_{m \subset R} \Delta NRM(t,t_m) \qquad (4)$$

where R is the set of relevant microblogs which are known from the feedback, NRM0 is the constant to initialize NRM. When substitute 11 and 12 into 13, we get

$$\gamma(t) = find\_kth(NRM_0 + \sum_{m \subset R}(\lambda\lambda^{\frac{-(t-t_m)^2}{2\varphi^2}} + 1)) \qquad (5)$$

Formula 5 shows how γ(t) addresses the one-side feedback. The threshold is γ(t) at time t. After a new relevant tweet arrives at time t', what samples under should be judged as positive to learn? γ(t') gives the samples posted near t' more probability to be judged as negative. Some examples located at the negative side are chosen to be learn, so the temporal microblog filter model have the ability to move the threshold toward negative side so that the one-side feedback problem is alleviated.

3.3. Estimating θ_q

A language model θ_q of query q indicates user interest. To resolve the problem of short query, we use the relevance language model [22], which is constructed on the background dataset, to expand the query. The relevance model deems that a query term is generated by the relevance model θ_R, which is derived from top ranked feedback microblogs by assuming them to be samples from the relevance model θ_R as follows:

$$P(w|\theta_R) \propto \sum_{m \in F} P(w|\theta_m) P(m|\theta_R) \tag{6}$$

where F denotes the feedback microblogs, usually chosen as the top-ranked retrieval microblogs for the query, $p(w|\theta_m)$ is the probability of the term w appearing in the microblog m, and the relevance model θ_R is approximated by the original query, thus we can obtain:

$$P(w|\theta_R) \propto \sum_{m \in F} P(w|\theta_m) P(\theta_m) \prod_{i=1}^{k} P(q_i|\theta_m) \tag{7}$$

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

$$P(w|\theta_q) = (1\text{-}\alpha\alpha)P(w|q + \alpha P(w|\theta_R) \tag{8}$$

where α is the interpolation weight, p(w|q) is the probability of word w seen in query q and θ_q is the language model of query.

3.4. Estimating θ_m

For the language model θ_m of microblog m, JM smoothing method is used to solve the zero probability problem[23] , which is given by:

$$p(w|\theta_m) = (1-\lambda)p(w|m) + \lambda p(w|C) \tag{9}$$

Where p(w|m) is the probability of word w seen in microblog m. The p(w|C) is the probability of w seen in collection C.

3.5. Estimating Sim (θ_q,θ_m)

Ponte and Croft[24] present a query likelihood scoring method to rank the documents based on the likelihood of the queries given by each document language model. In this scoring method, exploiting feedback documents to improve the ranking accuracy is difficult. So, we use Kullback-Leibler(KL) divergence scoring method[25] to overcome this problem by use the query language model and document language model based on the KL-divergence:

$$KL(\theta_q,\theta_m) = \sum_{w \in V} P(w|\theta_q) \log \frac{P(w|\theta_q)}{P(w|\theta_m)} \tag{10}$$

where $p(w|\theta_q)$ and $p(w|\theta_m)$ are the probabilities of w in θ_q and θ_m respectively.

4. Experiments

4.1. Datasets and Settings

To evaluate the properties of our model, we conduct experiments on dataset of the TREC 2012 Microblog Filtering Task. The corpus is comprised of 2 weeks of tweets sampled from Twitter. We downloaded 10,397,336 tweets by twitter crawler provided by track organizers. And total 3,754,077 tweets are remained after the following processing.

The main steps are as follows: (1) Retweets without "RT" are removed since they are to be judged as non-relevant. The retweets with"RT" are removed if there is no contents in front of "RT". But once there exist descriptions at the beginning of their tweet text, we only keep the words before"RT". (2) We filtered out all the non-English tweets by using language identifier tool provided by Nutch(http://nutch.apache.org). (3) Porter stemmer is used for stemming and stop words are filtered. The statistics of dataset are shown in Table 1.

Table 1. Statistics of Tweets in Dataset

# of Total Tweets	# of Retweets	# of Non-English Tweets	# of Tweets for Filtering
10, 397, 336	342, 652	6, 300, 607	3, 754, 077

Following the TREC 2012 microblog real-time filtering task, 48 queries are used in the experiment. The queries contain query words, the starting time and the ending time. The starting time points to the oldest known relevant tweet, and the ending time points to the newest one. The dataset is divided into three parts independently for every query. The three parts are as follows: (1) The tweets behind ending time are discarded. (2) The tweets prior to starting time are used as background training data to initialize filter and smooth language model. The given query and only one positive example, the first relevant tweet, are available for training. (3) The tweets between starting time and ending time are used as the filtering dataset. The filter is thereafter applied on each tweet to determine whether or not the tweet is relevant to the topic. The tweets, of which sim(θ_{URL},θ_q) is lower than the find_kth(1000) for the query, are removed.

In the TREC microblog real-time filtering task, 10 queries(1,6,11,16,21,26,31,36,41,46) form a training set and the remaining 38 queries form a testing set[6].

4.2. Measures

Following TREC 2012 microblog filtering tack, the filtering task is evaluated by using two main metrics: T11SU and $F_{0.5}$. Two other measures, precision and recall are reported for reference.

The T11SU measure adopted from TREC filtering is a linear utility function[6]. Imagine that the system receives a reward of two points for every relevant tweet which is judged relevant by filter, but takes a penalty of one point for every irrelevant tweet which is judged relevant. The utility is the total points scored[6]:

$$T11U=2*TP-FP \tag{11}$$

Where TP is the number of relevant tweets judged relevant, and FP is the total number of irrelevant tweets judged relevant. The linear utility function is equivalent to filter by estimated probability of relevance, in this case, to judge relevant if P (rel) > 1/3[6].

Utility values are unbounded, and hence need to be scaled to enable comparisons across topics. The utility scores are normalized to a fraction of their theoretical maximum, and scaled against an arbitrary minimum normalized utility value of -0.5 so that they may be averaged across queries[6]:

$$\mathrm{MaxU} = 2 * \mathrm{totalrelevant} \tag{12}$$

$$\mathrm{MinU} = -0.5 \tag{13}$$

$$\mathrm{NormU} = \mathrm{T11U/MaxU} \tag{14}$$

$$\mathrm{T11SU} = \frac{\max(\mathrm{NormU},\ \mathrm{MinNU}) - \mathrm{MinNU}}{1 - \mathrm{MinNU}} \tag{15}$$

A T11SU value of 1/3 can be achieved by a run that retrieves nothing not helping but not wasting the users time with irrelevant information either. This is called the “zero effort” baseline[6]. $F_{0.5}$ is defined based on F_β measure which is a function of precision and recall; the parameter gives the relative weighting of each component. B=0.5 gives an

emphasis to
precision. The F0.5 measure is computed as[6]:

$$F_\beta = (1+\beta^2)\frac{\text{precision} * \text{recall}}{\beta^2 * \text{precision} + \text{recall}} \tag{16}$$

4.3. Parameters Setting

In the experiments, the parameters of the language model are set according to the references. The parameters of temporal microblog filter model are trained by using the TREC microblog training data. The parameters in our experiments are set as follows:

Table 2. Parameters in Experiment

Formula	Parameter	Description	Reference
9	$\lambda = 0.5$	Language model JM smooth method	[23]
7	FBDC=20	feedback document count	[11]
7	FBTC=20	feedback term count	[11]
8	α=0.8	interpolation weight	[11]
5	NRM_0=1	Initial value of NRM	Optimized
5	λ =1	degree of influence	Optimized
5	φ=1200	influence extent	Optimized

4.4. Baselines

In our experiments, several methods are used as baselines.

LM represents a language model-based filter is reported in [11]. Rocchio represents a result of Rocchio applied on microblog filtering task, reported in Ref.[10]. LR is a filter using the logistic regression model. The result of SVM applied on this task can be seen in Ref.[12].

4.5. Experiment Results

The results and baselines on test data are reported in Table 3, in which TMFM represents the temporal microblog filtering model.

Table 3. Experiments Results

Model	T11SU	F0.5	Precision	Recall
LR	0.098	0.081	0.082	0.343
LM	0.332	0.201	0.386	0.099
SVM	0.322	0.158	0.195	0.221
Recchio	0.362	0.344	0.421	0.337
TMFM	0.409^{*+}	0.316^{*+}	0.392^{*+}	0.266^{+}

Note: * indicates the significant improvement over LR(paired t-test, $p < 0.01$), + indicates significant improvement over LM (paired t-test, $p < 0.05$).

It can be seen from Table 3 that the temporal microblog filter model (TMFM) has statistically significant improvements over several baselines.

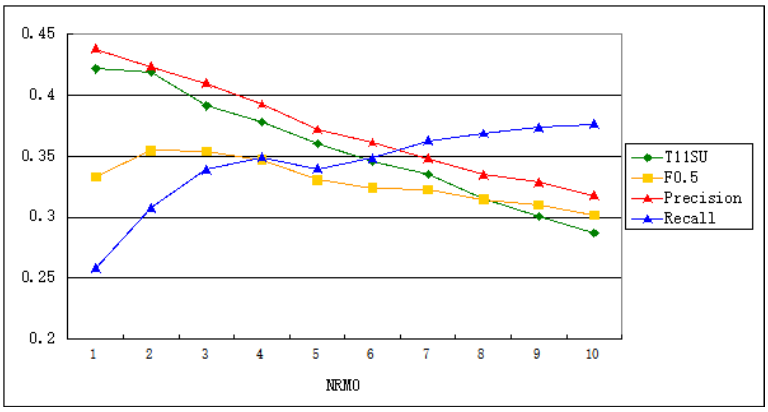

Figure 2. The Influence of NRM0

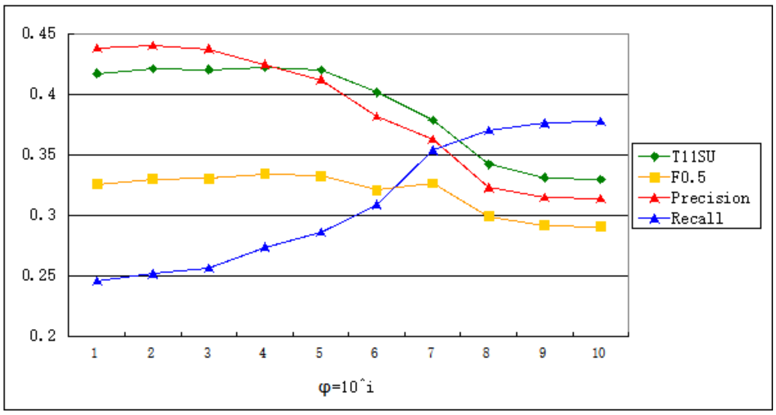

Figure 3. The Influence of φ

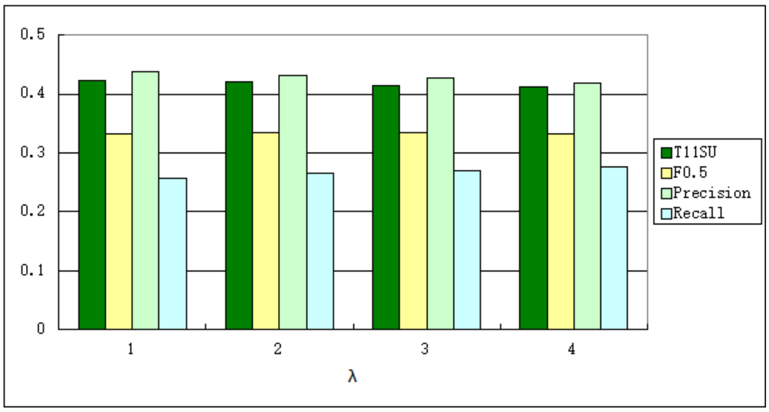

Figure 4. The Influence of λ

As shown in Figure 2, NRM_0 is the value of the initial NRM in formula 5. The higher initial value of NRM means the more relevant microblogs at the early stage. But in fact there are not so many relevant microblogs at the early stage, so the too high NRM_0 will lead to the descend performance.

From the Figure 3, we can find that φ decides the time extent which is influenced by the relevant microblogs. The recall continues to show an upward tendency with the precision sliding. When φ=5, it reaches the highest value at T11SU and $F_{0.5}$.

Figure 4 shows that higher value of λ, which decides the degree influenced by the relevant microblogs, attenuates the performance greatly.

5. Conclusions

This paper analyzes the task of microblog real-time filtering and interprets the filters based on retrieval model. And then, a temporal filtering model is proposed for addressing the one-side feedback problem by using temporal information. Although we are focusing on the filter based on language model, the proposed model can also be applied to other retrieval models.

We evaluated the hybrid model on the standard TREC real-time microblog filtering data sets. The performance is much better than the filter based on the language model and the logistic regression model, as well as SVM and Rocchio, even better than the best one in the TREC 2012 real-time microblog filtering track.

The real-time microblog filtering is a complicated problem and there are still large improvement space for our research. In the future, we plan to combine the classification model and retrieval model to improve the performance.

Acknowledgments

This work is supported by National Natural Science Foundation of China (Grant No. 61370170, No.61272384, No.61402134, No. 61173074) and Youth National Social Science Foundation of China (Grant No. 14CTQ032).

References

[1] H.Bosch, D.Thom, F. Heimerl, *et al*.. "Scatterblogs2: Real-time monitoring of microblog messages through user-guided filtering", IEEE Transactions on Visualization and Computer Graphics, vol.19, no.12, **(2013)**, pp.2022-2031.

[2] D. Wu, F. Yang, C. Zhang, "Statistical Methods based on Semantic Similarity of Topics Related to Microblogging", Journal of Software, vol.8, no.1, **(2013)**, pp.192-199.

[3] E. Diaz-Aviles,L. Drumond, L. Schmidt-Thieme, *et al*, "Real-time top-n recommendation in social streams", Proc. of the sixth ACM conference on Recommender systems, **(2012)** pp. 59-66.

[4] N. Asadi,J. Lin, "Fast candidate generation for real-time tweet search with Bloom filter chains", ACM Transactions on Information Systems (TOIS), vol.31, no.3, **(2013)**, pp.13:1-13:36.

[5] J. Golbeck, "The twitter mute button: a web filtering challenge", Proc. of the SIGCHI Conference on Human Factors in Computing Systems, ACM, **(2012)**, pp.2755-2758.

[6] I. Soboroff, I. Ounis, J. Lin, "Overview of the TREC-2012 microblog track", Proc. of the Twenty-First Text REtrieval Conference (TREC 2012), USA, **(2012)**.

[7] D. Sculley, "Practical learning from one-sided feedback", Proc. of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, USA, **(2007)**, pp.609-618.

[8] Y. Zhang, "Using bayesian priors to combine classifiers for adaptive filtering", Proc. of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, **(2004)**, pp.345-352.

[9] J. Allan. Incremental relevance feedback for information filtering. Proc. of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, **(1996)**, pp.270-278.

[10] N. Limsopatham, R. McCreadie, M. Albakour, *et al*., "University of Glasgow at TREC 2012: Experiments with Terrier in medical records", microblog, and web tracks. Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[11] Z. Han, X. Li, M. Yang, *et al*., "HIT at TREC 2012 microblog track", Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[12] F. Liang, R. Qiang, Y. Hong, *et al*.. PKUICST at TREC 2012 microblog track. Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[13] S. Karimi, J. Yin, P. Thomas, "Searching and filtering tweets: CSIRO at the TREC 2012 microblog track", Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[14] M. Efron , G. Golovchinsk, "Estimation methods for ranking recent information", Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. Beijing, China, **(2011)**, pp. 495-504.

[15] A. Dong, R. Zhang, P. Kolari, *et al*, "Time is of the essence: improving recency ranking using twitter data", Proceedings of the 19th international conference on World Wide Web. Raleigh, USA, **(2010)**, pp. 331-340.

[16] S. Cheng, A. Arvanitis, V. Hristidis, “How fresh do you want your search results?”, Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. San Francisco, USA ACM, **(2013)**, pp. 1271-1280.

[17] M. Keikha, S. Gi, C. Fabio, “ Time-based relevance models”, Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. Beijing, China, **(2011)**, pp. 1087-1088.

[18] J. Choi, W B Croft, “Temporal models for microblogs. Proceedings of the 21st ACM international conference on Information and knowledge management”, Maui, USA, **(2012)**, pp. 2491-2494.

[19] M. Albakour, C. Macdonald, I. Ounis, “On sparsity and drift for effective real-time filtering in microblogs”, Proc. of the 22nd ACM international conference on Conference on information \& knowledge management, ACM, **(2013)**, pp.419-428.

[20] K. Appel, L. Mathews, D. Lim, *et al*., “Siena's Twitter Information Retrieval System: The 2012 Microblog Track”, Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[21] A. S .E. Din, W. Magdy, “Web-based Pseudo Relevance Feedback for Microblog Retrieval”, Proc. of the Twenty-First Text REtrieval Conference, USA, **(2012)**.

[22] V. Lavrenko, W.B. Croft. Relevance based language models. Proc. of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, USA, **(2001)**, pp.120-127.

[23] C. Zhai, J. Lafferty, “A study of smoothing methods for language models applied to information retrieval”, ACM Transactions on Information Systems (TOIS), vol.22, no.2, **(2004)**, pp.179-214.

[24] J M Ponte J, W B Croft, “A language modeling approach to information retrieval”, Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM, **(1998)**, pp. 275-281.

[25] C. Zhai, J. Lafferty, “Model-based feedback in the language modeling approach to information retrieval”, Proc. of the tenth international conference on Information and knowledge management, ACM, **(2001)**, pp.403-410,

Authors

Zhongyuan Han, He was born in 1977, Ph.D. candidate, Associate Professor. His research interests include information retrieval, information filtering, and natural language processing.