Realtime Online Hot Topics Prediction in Sina Weibo for News Earlier Report

Conference Paper · March 2017

DOI: 10.1109/AINA.2017.66

4 authors, including:

Sha Yuan
Tsinghua University
12 PUBLICATIONS **39** CITATIONS
SEE PROFILE

Shuotian Bai
33
21 PUBLICATIONS **121** CITATIONS
SEE PROFILE

Realtime Online Hot Topics Prediction in Sina Weibo for News Earlier Report

Sha Yuan*, Zhe Tao†, Tingshao Zhu‡ and Shuotian Bai*§
*School of Information Engineering, Hubei University of Economics
†Department of Data Science, Beijing Juzi Technology Limited
‡Institute of Psychology, Chinese Academy of Sciences

***Abstract*—With the continuous growth of micro-blog services, Sina Weibo is increasingly found in the daily lives of ordinary Chinese individuals. More than one hundred million tweets are released in Sina Weibo everyday. By analyzing these mass data timely, media companies could learn how to generate buzz for new films, famous stars, or fashion shows more effectively. However, how to predict which topics will be the most popular search terms in Sina Weibo in realtime remains unknown. In this paper, we present a realtime hot topic prediction method in an online platform. Experiments are carried out on the platform to evaluate the proposed scheme. The results show that our model gets an average precision 44.32% and the median value is 45.83%. The proposed hot topic prediction method can predict the hot topics about 9.5 hours in average in advance.**

***Keywords*—Hot topic prediction; Sina Weibo; Topic extraction; Topic clustering**

I. Introduction

With the development of Internet and mobile phone technology, participation in social media has gradually become a routine part of many peoples' lives. Smartphones, which enable access to Internet services from anywhere at any time, accelerate the propagation of online information. As a result, the early prediction of hot topics is particularly important for online resources analysis and decision making [1]. Traditional researches managed to detect and identify the hotness of a topic through the search amount, the forward number, the reviews number and so on. The results could instantaneously recall the hot topics with a real-time detection system [2]. However, it cannot predict what topics will be hot in future.

Nowadays, social network sites become the main way to obtain the information. Sina Weibo [3], the Chinese version of Twitter, has more than 200 million registered users until May of 2016. Through Sina Weibo, users can get instant news and express attitudes. That is, Sina Weibo builds a network platform for information accessing and releasing. More than one hundred million micro-blogs are published in Sina Weibo everyday. The micro-blog terms in Sina Weibo are short texts. Each tweet is no more than 140 characters. Sina Weibo holds a hot topic list, in which there are 50 topics with the largest volume of searches. In essence, the list is a realtime hot topic detecting system, which provides an objective ranking on the hotness of topics.

Based on users' status contents and hot topic list of Sina Weibo, we try to quantify the current hotness of every topic, and predict its future tendency. To verify the results, we use the hot topic list as the evaluation criteria and predict whether the topic would be on the list in future. To achieve this, we collect and screen thousands of key opinion leaders (KOLs) as our information source, and constantly monitor the Weibo status they publish. The topic text in each Weibo status is extracted. And then, the statuses with the similar topic are merged into one cluster. Combined with the published time of each status, the changing tendency of topic hotness can be drawn by the sum of the comment count, the forward count and the likes count of its corresponding Weibo status. Setting different time granularity, the historical time series of topic hotness can be used to predict its future hotness and classify whether the topic would be on the hot topic list.

To this end, this paper presents a realtime online hot topic prediction model. Based on the Sina Weibo data of the KOLs, we propose an efficient topic extraction and clustering scheme. There are several difficulties in this process, such as data sparseness of short texts, massive data processing in real time, and various forms of Weibo status expression. Through extracting the static and dynamic features of the topics, we build the hot topic prediction model. The main contribution of this paper can be summarized as follows:

- We propose a topic extraction scheme based on the Sina Weibo textual data which is crawled in real time.
- Based on the topic extraction scheme, we apply the term-based vector space model (VSM) to construct a topic clustering scheme.
- We build a hot topic prediction model which can work on real-time environment. The results show that the proposed mechanism outperforms the conventional schemes in the aspect of hot topic

§Correspondence: No.8, Yangqiao Lake Avenue, Wuhan, China. E-mail: baishuotian@hbue.edu.cn.

prediction.

The proposed prediction model has been implement in the media company *"Juzi Entertainment"* [4] for hot topics earlier report. The editor will know the hot topics in Sina Weibo hot topic list 9.5 hours earlier by the prediction system. They will produce news articles in advance according to the predicted hot topics.

The rest of this paper is organized as follows. In section II, we outline the related work. Section III presents the hot topic prediction method. Section IV shows the experiment environment and the corresponding results that verifies the effectiveness of our approach. The relevant discussion is presented in Section V. Finally, the future work is discussed in Section VI.

II. Related Work

The United States Department of Defense Advanced Research Program (DARPA) [5] sponsored the Topic Detection and Tracking (TDT) [6] research to investigate the computational task in news broadcast programs, such as CNN news. The objective of the TDT program is to develop technologies that search, organize and structure multilingual, news oriented textual materials from a variety of broadcast news media [7]. The TDT problem may be divided into three major tasks: segmentation, detection and tracking [8]. The first step is to segment a stream of news data into distinct stories. The second step is to identify those news stories that are the first to discuss a new event occurring in the news. After giving a small number of sample stories about an event, the last step is to find all following stories in a news stream about the same event from some point in time onwards.

The existing studies mainly focus on detecting and tracking hot topics in a local area and during a particular period [9]. Data mining from trending topics have been applied to Twitter to summarize trending topics [10]. Ref. [11] proposes a novel hot topic detection scheme in Twitter. The basic approach is a classification method that mitigates the variation of posted words related to the same topic. However, due to semantic fluctuations, this classification does not work well in the aspect of hot topic detection. Ref. [12] finds out that the KOLs play a very important role in the propagation of hot topics. So that, they take the structure information and propagation characteristics of Sina Weibo as well as the users' influence into consideration in the hot topics discriminant model.

While the TDT has been studied, there has been little work done on predicting future information spreading patterns. Ref. [13] focuses on predicting information spreading in Twitter. They find that the most important features for future retweets prediction are the tweeter and retweeter. Ref. [14] outlines methodologies of detecting and identifying trending topics from Twitter's streaming data. However, the precision rate is relatively low. Ref. [15] presents an algorithm that can predict which topics will trend an average of an hour and a half before Twitter's algorithm puts them on the list. The training set they used consisted of 200 Twitter topics that did trend out and 200 that didnt. The training sets they used are very small. Besides, they set their algorithm loose on live tweets. So that, the algorithm is usefulness in the actual situation.

III. Hot Topics Prediction Method

To achieve efficient prediction in real time, we propose a hot topics prediction method in an online platform. Before modeling, there are three preprocessing steps including information source selection, topic extraction and topic clustering. The detailed flow chart is shown in Fig. 1.

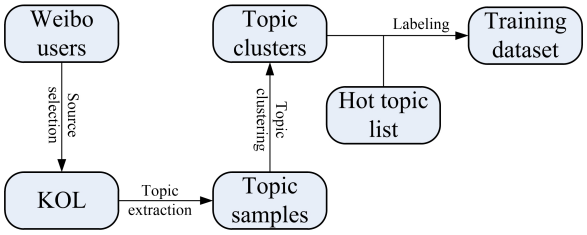

Fig. 1. Flow chart of data preprocessing

A. Information source selection

According to the report from CNNIC [16], Sina Weibo has more than 261 million active users. In the network topology of Sina Weibo, each user is an information source. Ideally, data monitoring should be carried out for each user. However, subject to the real time requirements, we need to screen out the KOLs. These users will be the information source of this research.

We deploy an online automatic crawling system in our computer cluster. According to the actual needs, the time interval of data crawling is set to 10 minutes. The system can crawl the web pages to download online data, and save it in the local database. 4433 topics in Sina Weibo hot topic list are analyzed during July 1, 2016 to July 31, 2016. For each topic, we statistics the distribution of its publishers before becoming a hot topic, and calculate hot topics publishing count for each user. Then, we rank the user in descending order with the count and retain the top 1500 users. The 1500 users are the KOLs of this research. We monitor and download their Weibo status in real time with a crawler system.

B. Topic extraction

Since the goal of this study is to predict the hot topic, the topic in the Weibo status need to be extracted firstly.

Before extracting topics, some preprocessing steps will be done on the Weibo status dataset. The specific rules are as follows:

1) Weibo statuses with less than 10 Chinese characters are removed.
2) The retweeted Weibo statuses are removed.
3) Non-text Weibo statuses, such as atlases, are removed.

According to the rules of Sina Weibo, the topic contents are denoted by identifiers "【···】" or "#···#", such as "【Topic field】Status field". Although "【···】" is Chinese punctuation, Sina Weibo supports both Chinese and English. Based on this rule, the following special situations will be encountered:

1) Some Weibo statuses with few word in topic field, such as "#Liying Zhao#", which are identified as names of superstars or places, should be removed. Therefore, the minimum length of the text in topic field is set as 4 Chinese characters to distinguish this situation.
2) Some topic field texts are meaningless, and cannot be used for topic generation, for example "#Good night#", "【Entertainment】". Such topics are the channel names of some organization's official Weibo account. In order to remove this kind of topics, we manually generate a stop topic list and filter the topics after extracting topic from Weibo status.
3) Some Weibo statuses contain multiple topic fields, as a result, topic extraction will get a number of topics, such as "【Topic field 1】#Topic field 2# Status field". In such a situation, we select the topic field with the maximum text length as the topic text.

In summary, based on the above processes, we can filter out the noisy sample of the original data, and extract the topic content from a Weibo status through recognizing topic identifiers.

C. Topic clustering

After the previous steps, topic content can be extracted from the Weibo status. For the hot topics, there are a number of Weibo statuses which have been reported. Therefore, we need to cluster the topics with same or familiar content into groups. To achieve this, the term frequency and inverse document frequency (TF-IDF) weight and VSM are used.

According to the corpus, the similarity between topic contents can be calculated based on the TF-IDF. We establish the topic vector of the sample topics, and calculate the vector distance between the sample topics in VSM. In this way, topic clusters can be set up. The information sources in this research generate massive new data of Weibo status, so that it is time-consuming to predict hot topics in the whole dataset. To solve this problem, an incremental computation strategy is used. The pseudo-code of the algorithm is shown as Algorithm 1.

Algorithm 1 *Incremental Topic Clustering Algorithm*

Initialize:
Weibo time step, wts;
Topic time step, tts.
Input:
Dataset of topics during tts, T;
Dataset of Weibo status with topic identifier during wts, DY;
Dataset of Weibo status without topic identifier during wts, DN.
Output:
The updated dataset of topics T;
Topic_Weibo relation dataset.
Steps:

```
for each dy ∈ DY
    topictemp = extract_topic(dy);
    for each t ∈ T
        if familiar(topictemp, t) == true
            update t
        else
            T.add(topictemp)
        end if
    end for
end for
for each dn ∈ DN
    for each t ∈ T
        if familiar(dn, t) == true
            update t
        end if
    end for
end for
```

Weibo time step is the time interval for awaking the algorithm. We set it as 10 minutes based on the data crawling rate and algorithm time complexity. That is, we extract topics of the new Weibo data in the 10 minutes, and update the existing topics or establish new topics. Topic time step is a threshold to judge whether a topic is old or not. We set it as 6 hours, that is, if the Weibo status of a topic has not been updated in the past 6 hours, this topic will be treated as an old topic and can be ignored for the following computation.

Therefore, the algorithm extracts the new published Weibo status in recent 10 minutes and divided Weibo status into two subsets according to whether the status has topic identifier or not. For each sample in dataset of Weibo status with topic identifier, DY, the algorithm extracts its topic field and gets a topic sample. If the topic sample is familiar with an existing topic t, then update the last updating time of topic t. Otherwise, establish a new topic using the topic sample. After the loop

of DY, the algorithm continues to loop each sample in dataset of Weibo status without topic identifier, DN. For each sample in DN, if the Weibo content is similar with an existing topic t, then update the last updating time of topic t. Otherwise, the algorithm will drop the Weibo status sample. Finally, the algorithm saves the dataset of updated topics and dataset of relationships between topics and Weibo statuses. Through our testing, the whole steps can be done in less than 3 minutes, which satisfies the requirement of real time.

D. Features

In order to predict the hotness of a topic, the following steps are needed. Firstly, we need to extract features of topics. Then, the hot topic classification model is built. Finally, we verify the performance of the model.

In this research, we design and extract two aspects of features: static features and dynamic features. The static features of topics includes statistics information and user attributes. Statistics information of a topic is the sum of the statistics information of its corresponding Weibo statuses, such as the count of likes, comments and forward. User attribute features includes the count of fans, Weibo status of the users and whether the user is verified. Dynamic features describe the interactive behavior of the fans in Weibo status. It is a time series features. We set different time granularity, 10 minutes, 20 minutes, 30 minutes, 60 minutes and 180 minutes, to dynamically extract the growth of the statistics information.

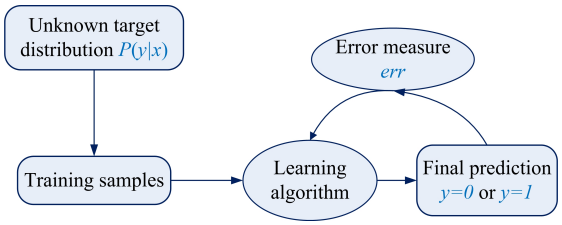

Fig. 2. Flow chart of the prediction method

E. Hot topic prediction

Hot topic prediction is a classification problem. The proposed prediction model is fitted with the realtime top trending searches of Sina Weibo. Before modeling, we use the hot topics in the realtime hot topic list as ground truth, and label the feature vectors of the extracted topics. In this study, the following "in future" is defined as two days. The following rules are used for labeling data:

- the topic turns to be in the hot topic list of Sina Weibo in future, label this feature vector as positive sample;
- the topic is not in the hot topic list of Sina Weibo in future, label it as negative sample.

After an un-interrupting feature extracting and labeling for one month, a training dataset with over 10000 samples is built.

Referring to our statistical data, there are about 100 hot topics among over 2700 un-hot topics every day. As a result, this is an unbalanced classification problem. Based on the aforementioned features, we build the hot topic prediction model with logistic regression algorithm [17]. With the selected parameters, the model can predict whether the topic will appear in the hot topic list in the next one hour. The detailed flow chart of the prediction method is shown in Fig. 2.

Let $T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_m, y_m)\}$ be the training set, in which $m = 10000$. The hypothesis function we used is

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}} \tag{1}$$

If $h_\theta(x) \geq 0.5$, the model predicts $y = 1$; otherwise, $y = 0$. Let $h_\theta(x)$ be the possibility when sample x is a positive sample, $1 - h_\theta(x)$ be the possibility when sample x is a negative sample.

The hot topic prediction is a supervised learning problem. We need to choose a function f in the hypothesis space F as the decision function [18]. For the given input x, $f(x)$ calculates the corresponding output y. The prediction value $f(x)$ maybe different from the actual value y. So that, we use a cost function for error correction. The cost function can be represented as

$$J(\theta) = \frac{1}{m} \sum_{m}^{i=1} Cost(h_\theta(x_i), y_i) \tag{2}$$

If we use the cost function of linear regression:

$$J(\theta) = \frac{1}{2}(h_\theta(x_i), y_i)^2 \tag{3}$$

in which, the $h_\theta(x)$ is coming from Eq. (1). In this way, $J(\theta)$ is non-convex. We may only find a local minimum value when we solve the non-convex function.

We need to find a cost function for our hot topic prediction model. From Eq. (1), we have

$$\ln \frac{h_\theta(x)}{1 - h_\theta(x)} = \theta^T x \tag{4}$$

where $h_\theta(x)$ can be regarded as the posterior probability $p(y = 1|x)$. So that, we obtain

$$\ln \frac{p(y = 1|x)}{p(y = 0|x)} = \theta^T x \tag{5}$$

From Eq. (5), we have

$$p(y = 1|x) = \frac{e^{\theta^T x}}{1 + e^{\theta^T x}} = \frac{1}{1 + e^{-\theta^T x}} = h_\theta(x) \tag{6}$$

and

$$p(y = 0|x) = \frac{1}{1 + e^{\theta^T x}} = 1 - h_\theta(x) \tag{7}$$

TABLE I
PRECISION EVALUATION OF THE PREDICTION MODEL

Items	Value	Prediction count	Accurate count	Occurrence date
Average precision	44.32%	24.07	10.67	–
Median precision	45.83%	24	11	Sep.1, 2016
Maximum precision	60.00%	20	3	Sep.12, 2016
Minimum precision	15.00%	20	3	Sep.3, 2016
Baseline precision	3.70%	about 100	about 2700	July, 2016

The parameter θ is calculated based on the principle of maximum likelihood estimate. The likelihood function of the prediction model is

$$l(\theta) = \prod_{i=1}^{m} p(y=1|x_i)^{y_i} p(y=0|x_i)^{1-y_i} \tag{8}$$

For convenience, we present the corresponding logarithmic likelihood function

$$l(\theta) = \sum_{i=1}^{m} [y_i \ln p(y=1|x_i) + (1-y_i) \ln p(y=0|x_i)] \tag{9}$$

From Eq. (9), we obtain

$$l(\theta) = \sum_{i=1}^{m} [y_i \ln h_\theta(x_i) + (1-y_i) \ln(1-h_\theta(x_i))] \tag{10}$$

Since $l(\theta)$ is a higher-order continuous differentiable convex function, we can solve it based on the gradient descent method according to the convex optimization theory [19]. When we get the maximum value of $l(\theta)$, the corresponding value of parameter θ is the most appropriate parameter estimation in the prediction model. Set

$$Cost(h_\theta(x), y) = \begin{cases} -\ln(h_\theta(x)) & if\ y=1 \\ -\ln(1-h_\theta(x)) & if\ y=0 \end{cases} \tag{11}$$

Combining Eq. (2) and Eq. (11), we know that the following cost function can be used in our prediction model.

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^{m} [y_i \ln h_\theta(x_i) + (1-y_i) \ln(1-h_\theta(x_i))] \tag{12}$$

That is,

$$J(\theta) = -\frac{1}{m} l(\theta) \tag{13}$$

Combined with the previous analysis, we just need to find the minimum $J(\theta)$ to get the most reasonable parameter estimation in the prediction model. In a word, the problem we need to solve in the hot topic prediction model is

$$\min \quad J(\theta) \tag{14}$$

Since the maximum value of $l(\theta)$ can be found based on the gradient descent method, we can get the minimum value of $J(\theta)$ at the same time.

The update rule based on the gradient descent algorithm [20] is

$$\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j} \tag{15}$$

It's going to do with the derivative of $J(\theta)$

$$\theta_j := \theta_j - \alpha \sum_{i=1}^{m} (h_\theta(x_i) - y_i) x_{ij} \tag{16}$$

All θ_j in the algorithm is updated simultaneously according to the Eq. (16).

IV. PERFORMANCE EVALUATION

Extensive online experiments have been performed to evaluate the proposed prediction mechanism.

A. Topic extraction and clustering

Topic extraction and clustering are the basic steps of our system, these operations are essential. However, there does not exist any objective automatic evaluation method. In this study, we invite two senior entertainment news editors from *"Juzi Entertainment"* to manually evaluate the accurate rate of topic extraction and topic clustering. For the exactly same dataset in July, 2016, we extract topics and cluster topics into groups using our algorithm. At the same time, the editors manually finish the work. Finally, we compare the results and calculate the precision and recall of topic extraction and the rand index precision of topic clustering. From the above testing strategy, the precision of topic extraction varies from 90% to 99%, the recall of topic extraction varies from 98% to 100%. The rand index precision of topic clustering varies from 80% to 95%.

B. Hot topic prediction

We use precision to evaluate the performance of the hot topic prediction model. For the topic we predict, there are following several situations:

1) The predicted topic has already been a hot topic. It means that our prediction is "late" compared with the hot topic list. For this situation, we eliminate the samples while evaluating the model.
2) The predicted topic turns into hot topic in future. For this situation, we treat the samples as correctly predicted cases.

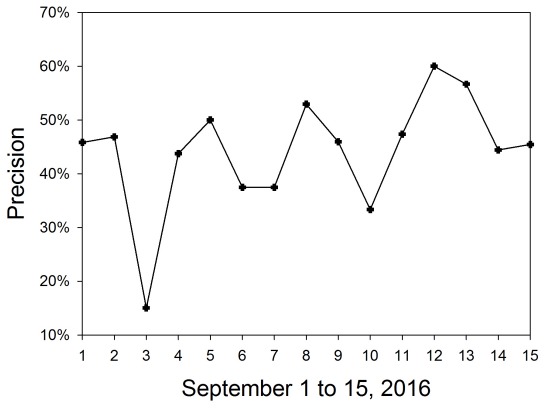

Fig. 3. Precision on real dataset during September 1 to 15, 2016

3) The predicted topic does not turn into hot topic in future. For these cases, we treat them as failure prediction samples.

Table I and Fig. 3 show the results on the real dataset during September 1, 2016 to September 15, 2016. In the table, prediction count is the count of the predicted topic not in hot topic list (the count of the above situations 1 and 3). The accurate prediction is situation 2, in which the predicted topic turns into hot topic in the Sina Weibo hot topic list in future. The baseline of the prediction precision is $100/2700 = 3.7\%$. During our testing time interval, the hot topic prediction model makes 361 times prediction totally, and 160 times are correct among them. Therefore, our model gets an average precision 44.32% and the median value is 45.83%. For September 12, our model gets a local extreme value 60%, while our model works badly on September 3 with only 15% precision. The average result has more than doubled the previous work in Ref. [14].

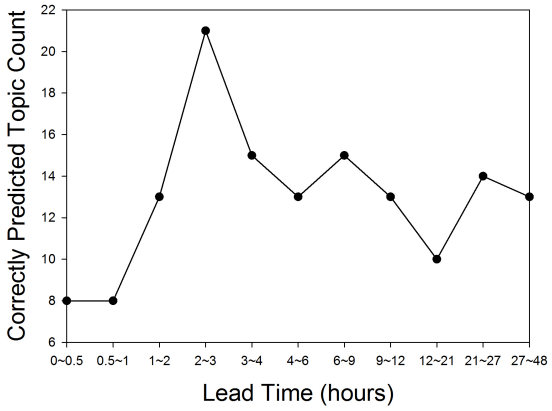

Fig. 4. Lead time distribution of correctly predicted topics

C. The time ahead of correctly predicted topics

Our objective is to early predict hot topics in Sina Weibo for news earlier report. Therefore, for the correctly predicted samples, the time ahead quantity will be calculated. Fig. 4 and Table II show the time ahead distribution of the correctly predicted topics in our test dataset. From the results, we find that the proposed model can forecast hot topics in advance, and the time ranges from 5 minutes to 2693 minutes (about 1 day and 21 hours). The average ahead quantity is 568 minutes (about 9.5 hours). The time head of most samples ranges from 2 hours to 3 hours.

TABLE II
TIME AHEAD OF CORRECTLY PREDICTED TOPICS

Items	Quantity(minute)
Maximum ahead	2693
Minimum ahead	5
Median ahead	291
Average ahead	568
Mode ahead	[120, 180]

V. DISCUSSION

In this study, we predict hot topics based on Weibo social network site platform. The proposed prediction method is termed as "*ROHTP*". On our testing dataset, we get an average predicting precision 44.32%, which is an excellent results corresponding to prediction baseline and topic prediction amount. What's more, the results have more than doubled the previous work. The comparison results are shown in Fig. 5, in which "*STD1*" and "*STD2*" are the results in Ref. [14].

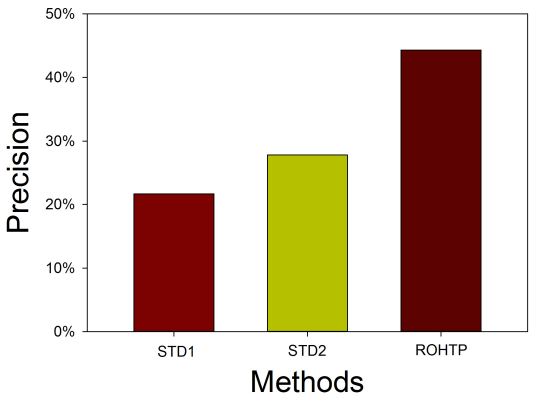

Fig. 5. Lead time distribution of correctly predicted topics

For the correctly predicted topics, our model can recognize them 568 minutes (about 9.5 hours) ahead of its appearance in the hot topic list of Sina Weibo. In general, it costs about one hour for an editor to produce a piece of news article which is significantly

less than our time ahead. As a result, combined with their experiences in journalism, our model can assist news editors to early report hot topics.

For the wrongly recalled topics, we can still find some interesting patterns. They can be recalled, because their features satisfy our model, which means that the topics have many comments, forward or likes. They do not appear in hot list because the topics are politics related. We can imagine that the hot topic list of Sina Weibo is controlled by Sina Weibo editors manually. As far as our observation and statistics, topics of politics, malignant event, and foreign events are possible to be filtered by the editors of Sina Weibo.

VI. Future Work

In the future work, we will continue to improve our prediction method. The information sources will be dynamically changed in the process of prediction. If some KOLs don't publish tweets for a period, they will be removed from the sources. At the same time, some new hot topic publishing users will be added into our sources. In the next step, the topic extraction and clustering will be worked on the Sina Weibo environment. The prediction algorithm will also be studies to improve its performance. What's more, we will try to distinguish the politics related topics that are removed by Sina Weibo to raise our precision.

Acknowledgment

This work is supported by the Natural Science Foundation of Hubei Province (2016CFB208), the Project (13Q100) from the Education Department of Hubei Province, Srategic Priority Program (X-DA06030800), 100-Talent Project (Y2CX093006) from Chinese Academy of Sciences.

References

[1] P. Bao, H. W. Shen, J. Huang, and X. Cheng, "Popularity prediction in microblogging network: A case study on sina weibo," pp. 177–178, 2013.

[2] H. F. Ma, Y. X. Sun, M. H. Z. Jia, and Z. C. Zhang, "Microblog hot topic detection based on topic model using term correlation matrix," in *2014 International Conference on Machine Learning and Cybernetics*, 2015, pp. 126–130.

[3] Q. Gao, F. Abel, G. J. Houben, and Y. Yu, "A comparative study of users' microblogging behavior on sina weibo and twitter," in *International Conference on User Modeling, Adaptation, and Personalization*, 2012, pp. 88–101.

[4] HAPPYJUZI, "Juzi entertainment," Website, http://www.happyjuzi.com/.

[5] WIKIPEDIA, "Darpa," Website, https://en.wikipedia.org/wiki/DARPA.

[6] V. Strugala, J. Avis, L. M. Johnstone, I. G. Jolliffe, and P. W. Dettmar, *Introduction to Topic Detection and Tracking*. Springer US, 2002.

[7] J. G. Fiscus and G. R. Doddington, *Topic detection and tracking evaluation overview*. Springer US, 2002.

[8] J. Carthy and A. F. Smeaton, "The design of a topic tracking system," *Proceedings of Annual Colloquium on Information Retrieval Research*, pp. 84–93, 2000.

[9] K. Kamaldeep and G. Vishal, "A survey of topic tracking techniques," *International Journal of Advanced Research in Computer Science and Software Engineering*, pp. 384–393, 2012.

[10] B. P. Sharifi, D. I. Inouye, and J. K. Kalita, "Summarization of twitter microblogs," *Computer Journal*, vol. 57, no. 3, pp. 378–402, 2013.

[11] S. Ishikawa, Y. Arakawa, S. Tagashira, and A. Fukuda, "Hot topic detection in local areas using twitter and wikipedia," in *Arcs Workshops*, 2012, pp. 1–5.

[12] D. Li, Y. Zhang, X. Chen, L. Cao, C. Zhou, D. Li, Y. Zhang, X. Chen, L. Cao, and C. Zhou, "Detecting hot topics in sina weibo based on opinion leaders," *ccit-14*, 2014.

[13] T. R. Zaman, R. Herbrich, J. V. Gael, and D. Stern, "Predicting information spreading in twitter," *Computational Social Science and the Wisdom of Crowds Workshop*, 2010.

[14] J. Benhardus and J. Kalita, "Streaming trend detection in twitter," *International Journal of Web Based Communities*, vol. 9, no. 1, pp. 122–139, 2013.

[15] M. N. Office, "Predicting what topics will trend on twitter," *Communications of the Acm*, 2012.

[16] CNNIC, "China internet network information center," Website, http://www.cnnic.net.cn/.

[17] S. Menard, "Applied logistic regression analysis," *Technometrics*, vol. 38, no. 2, pp. 184–186, 2010.

[18] V. N. Vapnik, "Statistical learning theory," *Encyclopedia of the Sciences of Learning*, vol. 41, no. 4, pp. 3185–3185, 2010.

[19] Boyd, Vandenberghe, and Faybusovich, "Convex optimization," *IEEE Transactions on Automatic Control*, vol. 51, no. 11, pp. 1859–1859, 2006.

[20] A. zilinskas, "Practical mathematical optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms," *Thesis*, no. 6, pp. 613–615, 1993.