Journal of Computers Vol. 30 No. 1, 2019, pp. 42-51
doi:10.3966/199115992019023001005

Microblog User Interest Mining Based on Improved TextRank Model

Rui Niu[1], Bo Shen[2*]

[1] Department of Electronic and Information Engineering, Beijing Jiaotong University Beijing, China
wood1s@163.com

[2] Department of Electronic and Information Engineering, Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing, China
bshen@bjtu.edu.cn

Received 9 January 2018; Revised 21 January 2018; Accepted 22 January 2018

Abstract. As microblogs have become one of the most important social platforms, it is considered to be extremely valuable to extract user interests hidden behind microblogs. In this paper, we introduce a framework, which is built on the improved TextRank model, to analyze the personal interest of microblog users. In the framework, we first create a catalog of user interests basing on hot tags of Sina Weibo, the largest microblog system in China. And then TF-IDF factor is used in TextRank model to deal with pre-processed microblog contents. After ranking and mapping all extracted words into user interests catalog established previously, we get corresponding user interests tags and a user interests model. Experimental results on Sina Weibo data imply that the proposed framework outperforms other existing methods.

Keywords: microblog, TextRank, TF-IDF, user interests

1 Introduction

By August 2017, Sina Weibo has obtained 360 million monthly active users. Through Sina Weibo, users can post microblogs with less than 140 characters, follow other users that they are interested in to share their personal life, discover interesting contents, and pay close attention to hot topics. It is not surprising that there is enormous amount of valuable information hidden behind microblogs such as user interests, hot topics, and etc. Therefore, how to extract user interests efficiently and customize contents based on user interests effectively have become essential topics of the area.

Related research over user interest extraction are either based on behavior analysis or text/content analysis. Different from typical text/content analysis, contents analysis of microblogs is much more challenging because microblog texts are sparse and noisy due to the limitations on the number of characters and the nature of social media. Some scholars have expanded texts from semantic perspective through external knowledge bases (such as HowNet Knowledge Base and Wikipedia) to broaden the content of texts, which helps to deal with the sparsity of short texts at some level [3-4]. Scholars such as Yan, however, expanded texts through Biterm Topic Model, which utilizes new terms that have close meanings with the targeted keywords in short texts to respond to the sparsity of short texts [5]. Scholars like Kim use the term frequency (TF) distribution and “Like” functions on Facebook to extract Facebook users’ interests and preferences [6]. Literature [7] and [8] further study user interests and preferences based on the contents produced by users, while in literature [9], Chen and his colleagues build Bag-of-Words style user profile model based on TF-IDF ranking of terms in users posts. Term frequency ranking models like TF-IDF only considers the importance of single terms in texts from probability aspects when studying keywords of user interests. However, it ignores the correspondence between term to term from semantic aspects. In literature [9], Zhao [10] modified LDA (Latent Dirichlet Allocation) model [11]

[*] Corresponding Author

specifically against the characteristics of short texts in Twitter and further came up with Hashtag-LDA model, which completed the task of topic recommendation based on user preferences. Last but not least, Wu and his colleagues [12] extract and recognize the keywords of Twitter user interests through TF-IDF [1] and TextRank [2].

In this paper, we first analyze text content, and then propose a new framework to extract interests of microblog users:

- Create user interests catalog according to hot topic tags of Sina Weibo by ICTCLAS2016.
- Embed TF-IDF factor into the TextRank model to construct a new method of extracting user interest candidate terms.
- Map candidate terms into user interests catalog to obtain the keywords that are able to represent user interests and preferences.

2 Microblog User Behavior and Content Analysis

2.1 Microblog User Behavior Analysis

Java et al. [13] classified microblog users into three types: hot spots generating users, hot spots spreading users and ordinary users. The ordinary users is the main component that constitutes a microblog user group. In order to accurately describe user preferences, it is necessary to analyze microblog user behavior. In the personalized recommendation, the user's main behavior can generally be divided into two kinds, named the explicit behavior and the implicit behavior.

Explicit behavior refers to the user's explicit expression of interest in an item. In social network environments, for example, the user likes a microblog and forwards the microblog, which are all explicit behaviors. Implicit behaviors are those user interest behavior that are not explicitly and directly expressed, such as browsing, collecting and so on.

In microblog, the user behaviors are mainly in the following forms:

(1) Posting microblog;
(2) Forwarding other user's microblog ;
(3) Commenting other user's microblog;
(4) Liking other user's microblog;
(5) Collecting other user's microblog;
(6) Deleting microblog;
(7) Reporting other user's microblog;

Microblog user behaviors are the direct manifestation of their interests, preferences and concerns. In all these behaviors, 1, 2 not only shows the user's favorite of the content, but also want other users to like it, so the interest is more intent. 3, 4, 5 shows the interest of the content, but the interest is not strong enough to promote it.

The difference between behaviors of microblog users reflects different preferences of users for content. In the process of user preference analysis, the corresponding content weights need to be adjusted according to different user behaviors so as to more accurately model user preferences.

Due to the shorter length of microblog, using a single microblog as document will lead to a corpus with a strong sparsity, therefore we combine microblogs by certain rules as a document to expand text length.

2.2 Microblog Content Feature Analysis

Short text length. Because microblog is mainly for the convenience of users to quickly and easily share their daily life and record the mood attitude, the text length is limited to 140 words. Comparing to ordinary text analysis, the content length is shorter. In fact, microblog users tend to be briefer, only a dozen or a few words when they post microblog. This leads to a relatively strong sparseness.

Diversified content structure with noise. Microblog content is more popular and life-oriented, in which user's expression has a strong randomness. Meanwhile, microblog content also has strong colloquial, mixed with a large number of emoticons, abbreviations, homophonic and network terms, as well as web links, pictures, videos and other elements. For the task of extracting keywords, these are all noise.

Content is time-sensitive. More and more users regard the microblog as a tool of paying attention to current affairs, news and of recording the current life. The content on the microblog system has strong timeliness. And the change of subjects implied by microblog contents is the variation of user interests.

3 TextRank Model with TF-IDF Factors

3.1 User Interests Catalog

As microblog contents are colloquial and cluttered, user interest tags extracted directly from microblog contents might lead to unspecified and imprecise results, which will not reflect true user interests and preferences. By building knowledge base, mapping and classifying the knowledge base with user interest tags, we will be able to understand and convey user interests better.

We take classification entries under the "hot posts" of Sina Weibo as root directory, and use entries under the root directory to search for related contents in Weibo. Next, utilizing the keywords extraction function of ICTCLAS2016 to process the results obtained through previous step, the experiment now has acquired the secondary directory. Then, the term correspondence analysis function of ICTCLAS2016 will be able to analyze, clean and organize the data to obtain interest correlated terms corresponding to the secondary directory.

Table 1. User interests catalog

Root Directory	Secondary Directory	Interest Correlated Terms
Sports	Soccer	RealMadrid; Messi; Guoan; Goal; Score; …
	Basketball	Playoff; Champion; Goal; Winner; …
	…	…
Electronics	Cell Phone	iPhone; Samsung; Battery; RAM; …
	Computers	Lenovo; Configuration; New Release; Game; …

3.2 Modified TF-IDF algorithm

TF-IDF is a very typical statistic algorithm to evaluate the importance of a term in the document. TF-IDF is the product of Term Frequency (TF) and Inverse Document Frequency (IDF). The importance of a term increases as the appearance frequency of the term in the document increases, and it also increases as the number of documents that contain the term in the corpus increase.

Normally, user interest is not immutable. As time shifts, user interest may also be offset. Therefore, for the keyword extraction of text posted by the user, it is not realistic to give the same weight to the latest published microblog text and the release earlier text. In order to simulate the transfer of user interest over time, we adjust the TF-IDF algorithm according to the time characteristic of microblog and introduce a TF-IDF timing factor based on the interest offset of users.

Microblog users can also forward the microblogs posted by other users and comment on them, besides posting microblog directly. These various behaviors also reflect user's preferences for the corresponding content to some extent. Therefore, user behavior also need to be considered in keywords extraction. Here we assume that the original microblog posted by a user is more reflective of user preference than its forwarding and comments. So, when calculating TF-IDF, different weights are set for different user behaviors to simulate the importance of interests reflected by different user behaviors in the actual situation.

Define User Interest Catalog $C=\{l_1,l_2,...,l_i,...,l_n\}$, n is the total number of secondary directories, $1\le i\le n$. For any $1\le i\le n$, l_i contains several correlated terms $\{l_{i1},l_{i2},...,l_{ij},...,l_{im}\}$, $1\le j\le m$. TF-IDF value of a certain term can be calculated by following formula:

$$W_{TF-IDF}(l_{ij},u)=t_1[\mu_1 W_{TF-IDF}(l_{ij},d_{11})+\mu_2 W_{TF-IDF}(l_{ij},d_{12})]+t_2[\mu_1 W_{TF-IDF}(l_{ij},d_{21})+\mu_2 W_{TF-IDF}(l_{ij},d_{22})] \quad \textbf{(1)}$$

where d_{11} is the original contents posted by user u in the past month. d_{12} is the microblog contents reposted or commented by user u in the past month, d_{21} and d_{12} are the corresponding contents posted by user u. Let t be the time. As interests of user u change as time goes on, microblog posted by a user recently is a better reflection of present user interests in comparison to the contents posted by the user prior to certain time period. And at the same time, original post is a better reflection of user interests in comparison of reposted and commented microblog contents. Therefore, let t_1=0.7 , t_2=0.3 , μ_1=0.6 , μ_2=0.4 .

$$W_{TF-IDF}(l_{ij},d_{pk}) = W_{TF}(l_{ij},d_{pk})W_{IDF}(l_{ij},d_{pk}); p,k \in \{1,2\} \quad \textbf{(2)}$$

$$W_{TF}(l_{ij},d_{pk}) = \frac{c(l_{ij})}{c(d_{pk})}; p,k \in \{1,2\} \quad \textbf{(3)}$$

where $c(l_{ij})$ is the number of appearance of term l_{ij} in document d_{pk}, and $c(d_{pk})$ is the total number of terms in document d_{pk} .

$$W_{IDF}(l_{ij},d_{pk}) = \lg\frac{|d_{pk}|}{|d_{l_{pk}}|}; p,k \in \{1,2\} \quad \textbf{(4)}$$

where $|d_{pk}|$ is the total number of microblogs posted with the same behaviors within corresponding time period, and $|d_{pk}|$ is the total number of microblogs posted that contain term l_{ij}

3.3 TextRank

TextRank is inspired by Google PageRank Algorithm. It is a text keywords ranking algorithm based on graphic model. TextRank transforms texts into graphic model of terms, and utilize the co-occurrence relationship between terms to rank keywords in the texts. Different from models like LDA, TextRank does not need training and studies over the corpus, and it is able to extract keywords from just one single piece of document.

TextRank model can be represented by a directed graph $G = \{V, E\}$, and this experiment utilizes sliding window algorithm to obtain relationships between terms [14]. Terms on the left side of Target term in the window is the out-degree of the graph, while terms on the right side of target term is the in-degree of the graph. $V = \{l_1, l_2, ..., l_{|V|}\}$ is a set for all points in the graph, and the line from point l_i to point l_j is $E = \{(l_i, l_j) : 1 \le i, j \le |V|\}$. For given point l_i , the formula below can be used to calculate the TextRank weight at this given point

$$S_{TextRank}(l_i) = (1-d) + d \times \sum_{l_j \in in(l_i)} \frac{w_{ji}}{\sum_{l_k \in out(l_j)} w_{jk}} S_{TextRank}(l_j) \quad \textbf{(5)}$$

In this formula, w_{ij} indicates the in-degree from point l_j to point l_i , and d is damping coefficient, which is generally set to be 0.85. Besides, the size of this window is set to be 5.

3.4 TextRank with TF-IDF factors

TF-IDF method only considers the importance of terms reflected by term frequencies from statistic perspectives, but it ignores the correspondence between terms, and does not evaluate the importance of terms reflected from semantic perspective. TextRank model is just the opposite. It reflects the correspondence between terms, and extracts keywords through the voting relationship of terms without taking the term frequency to count.

When iterating in the TextRank model, the probability of jumping randomly from one term to another term is 1-d, which caused the probability of jumping to a low correlated term and the probability of jumping to a high correlated term is equal, or even higher. Therefore, embedding TF-IDF factor into TextRank model helps the algorithm to jump according to the importance of terms, and increase the probability of jumping to a high correlated term. The modified TextRank formula is following:

$$S_{TextRankm}(l_i,u)=(1-d)+d(1+W_{TF-IDF}(l_{ij},u))\times\sum_{l_j\in in(l_i)}\frac{w_{ji}}{\sum_{l_k\in out(l_j)}w_{jk}}S_{TextRank}(l_j) \tag{6}$$

In Formula 6 w_{ij} indicates the in-degree from point l_j to point l_i , d is damping coefficient, which is generally set to be 0.85 and the size of window is set to be 5. $W_{TF-IDF}(l_{ij},u)$ in this formula represent the TF-IDF value of corresponding term. Input data to the loop iteration and wait until convergence. Then take the top-N terms with highest values as candidate interest correlated terms.

This paper uses loops to calculate the TF-IDF value of candidate interest correlated terms in the user interests catalog that are contained in users' microblogs, and the algorithm is following:

Algorithm 1. TF-IDF

```
Input: Pre-processed microblog documents: $D=\{d_{pk}\};p,k\in\{1,2\}$
        Candidate Interest Correlated Term Set: $C=\{l_1,l_2,...,l_m\}$
Output: TF-IDF values of candidate terms in the user interests catalog: $S_{TF-IDF}(C,D)$
initialization: $S_{TF-IDF}(C,D)\leftarrow\{\}$
for $k\leftarrow 1$ to $2$
  for $i\leftarrow 1$ to $m$ do
    for $j\leftarrow 1$ to $|l_i|$ do
      $W_{TF-IDF}(l_{ij},d_{pk})=\frac{c(l_{ij})}{c(d_{pk})}\lg\frac{|d_{pk}|}{|d_{l_{pk}}|}$
    end for
    $S_{TF-IDF}(C,d_{pk})\leftarrow S_{TF-IDF}(C,d_{pk})\cup W_{TF-IDF}(l_{ij},d_{pk})$
  end for
  $S_{TF-IDF}(C,D)=S_{TF-IDF}(C,D)\cup S_{TF-IDF}(C,d_{pk})$
    end for
return $S_{TF-IDF}(C,D)$
```

Take the interest correlated terms and its TF-IDF values obtained through algorithm one as input of the modified TextRank model, start loop iteration until the results converge and form user eigenvectors. Modified TextRank Algorithm is as following:

Algorithm 2. Modified TextRank Model

```
Input: Pre-processed microblog documents: $D=\{d_{pk}\};p,k\in\{1,2\}$
        Interest Correlated Term Set: $C=\{l_1,l_2,...,l_m\}$
Output: Interest feature vectors $V$
Initialization: $d$=0.85
$V\leftarrow\{\}$
repeat
  for $i\leftarrow 1$ to $m$ do
    for $j\leftarrow 1$ to $|l_i|$ do
      $S_{TextRankm}(l_i,u)=(1-d)+d(1+S_{TF-IDF}(l_{ij},u))\times\sum_{l_j\in in(l_i)}\frac{w_{ji}}{\sum_{l_k\in out(l_j)}w_{jk}}S_{TextRank}(l_j)$
```

```
        end for
    end for
until $S_{mTextRank}(l_i)$ converge
    for $i \leftarrow 1$ to $m$ do
        for $j \leftarrow 1$ to $|l_i|$ do
            $w_i += S_{TextRankm}(l_i, u)$
        end for
        $V \leftarrow V \cup l_i : w_i$
    end for
return $V$
```

Based on $S_{TextRankm}(l_i,u)$, we are able to calculate the $S_{TextRankm}$ value of candidate interest correlated terms in the posts of microblog users, rank all the interest correlated terms, select the top-N of them, and find their corresponding root and secondary interest tags from the user interest catalog to set as the final user interest tags.

3.5 Interest Tag Filtering

While finding primary and secondary interest tags based on candidate interest correlated terms, it is possible that the same candidate term maps into multiple interest tags. For instance, “score” is a tag under both user interests catalog of soccer and basketball. Therefore, this paper proposes trigger term weighting method based on the actual situation where one interest tag can be described by multiple terms, and contents that users are more interested in generally has higher frequency to appear.

For candidate term l that is existing under multiple primary and secondary interest tags, we will find out the corresponding primary and secondary interest tags l_i of the user’s other candidate interest terms. If only 1 on the user’s interest tags l_i contains the candidate term l, then the corresponding interest tag for the candidate term l is l_i. If there are multiple interest tags contain the term l, then calculate the sum of TextRank value of corresponding candidate terms for each tag, and then the interest tag for the term l will be the l_i that has the largest sum. However, if none of the user’s interest tags contain the term l, then select the tag randomly based on the probability of l mapping into n tags, which is $1/n$.

4 Experiment Results and Analysis

4.1 Experiment Procedures

Fig. 1 shows the framework of mining microblog user interests, which is used to match user interest subjects and keywords:

(1) From Sina Weibo API randomly crawl and store user-related microblogs as experiment data.

(2) Use Sina Weibo Hot Weibo Category to build user interest categories.

(3) Preprocess obtained microblog.

(4) Calculate the TF-IDF weight for Weibo content.

(5) Extract interest related key words.

(6) Use interest related key words to match user interest categories and finally generate user interest subjects.

4.2 Experiment Data

This experiment crawls microblog contents posted by 1500 Sina Weibo users between March 2017 to June 2017 through Sina Weibo API, filters out users that posted less than 30 microblogs and then uses 361852 microblog posts from 1175 users as experiment data. Among these microblog posts, 173166 of them are original posts, 60209 of them are reposts, and 128477 of them are comments.

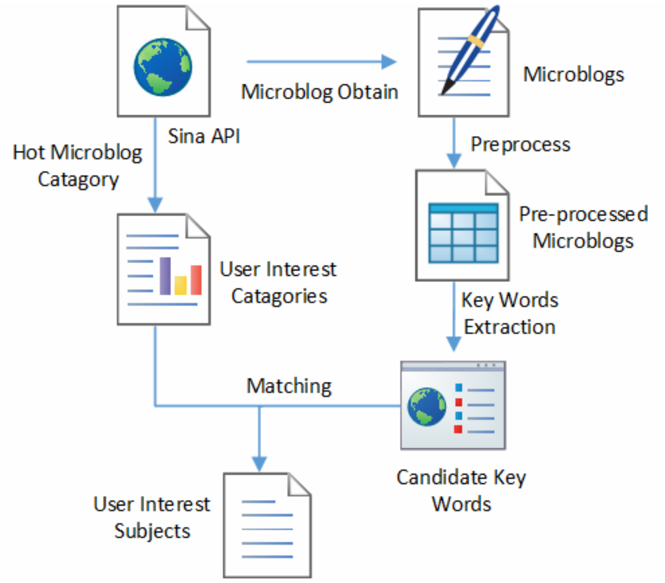

Fig. 1. The procedures of generating user interest subjects

The experiment aggregates the microblog posts posted by the same user into one document. And for the aggregated user texts, here are the steps to process data: 1. Texts cleaning: filter and delete the special characters in the microblogs, such as emoji, punctuations like "@" and "//", and terms that are no longer used. 2. Term Parting: Part and identify the part of speech of terms in the microblog texts through related functions of ICTCLAS2016, filter the results and only keep nouns and verbs.

4.3 Candidate Interest Terms Extraction Results

Headings. Take secondary tag "cell phone" as example, partial results of interest correlated terms extraction are as Table 2.

Table 2. Candidate interest terms

Candidates	TF/IDF	Candidates	TF/IDF
Flash Memory	6.51/4	Release	5.43/3
Huawei	6.67/3	Review	5.21/10
iPhone	6.31/21	Meizu	4.59/20
Snapdragon	5.92/11	Samsung	4.41/5
Xiaomi	6.11/32	Touch	4.33/7

4.4 Results Analysis

4.4.1 Accuracy

Accuracy is calculated based on following formula:

$$R_{precision} = \frac{N_1}{N_1 + N_2} \tag{7}$$

In the formula, N_1 is candidate interest terms, N_2 is filtered un-related terms, and the top-N accuracy of TF-IDF, TextRank, TextRankm are listed Table 3.

Table 3. Accuracy

Methods	Top-5	Top-10	Top-15
TF-IDF	0.511	0.451	0.404
TextRank	0.469	0.410	0.396
TextRankm	0.521	0.472	0.443

According to the table above, the modified TextRankm model performs better at top-5, top-10, and top-15 in comparison with TF-IDF and TextRank. Compare to TF-IDF, the accuracy of TextRankm is 0.01, 0.021, and 0.039 higher, and compare to TextRank, the accuracy of TextRankm is 0.052, 0.062, and 0.047 higher.

4.4.2 Average Accuracy

Formula to calculate average accuracy is as following:

$$R_{mp} = \frac{\sum_{i=1}^{N} S(i)}{N} \tag{8}$$

where $S(i)$ is the value of the ith candidate term, and N is the total number of candidates. The top-N average accuracy of TF-IDF, TextRank, and TextRankm are listed in Fig. 2.

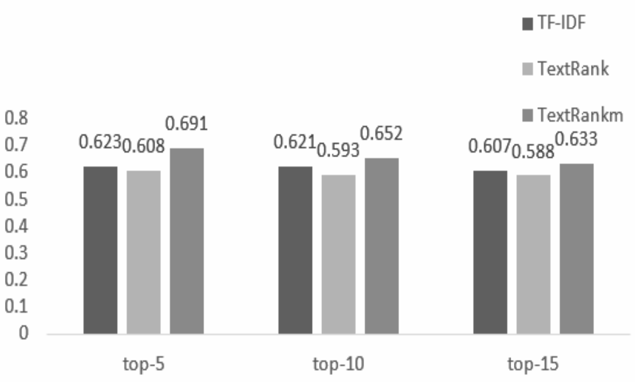

Fig. 2. Average accuracy

According to Fig. 1, average accuracy of TextRankm is higher than average accuracy TextRank, and TF-IDF. The top-5, top-10 and top-15 average accuracy of TextRankm is 0.691. 0.652, 0.633, which is 0.029, 0.030, 0.021higher than TF-IDF, and 0.083, 0.059, 0.023 higher than TextRank.

4.4.3 Recall Rate

Recall Rate is calculated using following formula:

$$R_{recall} = \frac{N^{'}}{N} \tag{9}$$

Where *N'* is selected candidate interest term, and *N* is the total number of candidate interest terms. The corresponding recall rate of three methods are illustrated as Fig. 3.

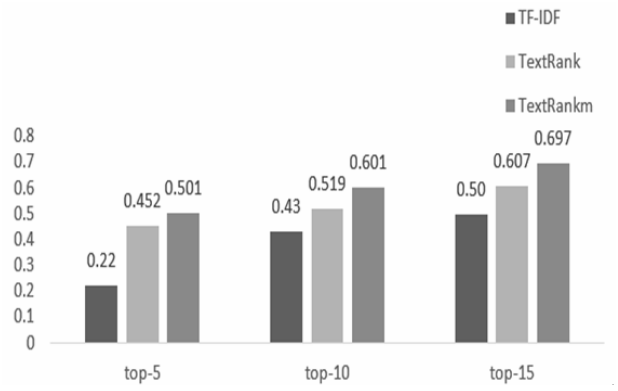

Fig. 3. Recall Rate

As the texts of microblogs are relatively short, recall rate of TF-IDF is lower. According to Fig. 2, recall rate of TextRankm is relatively high.

5 Conclusion

Experiment in this paper proposes a model to extract user interests and preferences from microblogs. It builds up a user interests catalog, modifies TF-IDF method based on characteristics of microblog users' behaviors, and imports TF-IDF into TextRank model as a factor. Data processing results of the modified methods indicates that the methods proposed in this paper performs better than TF-IDF model and TextRank model.

However, this model only mines the text contents of microblogs without taking user social networks, personal features, and other factors into consideration. Plus, the user interests catalog is yet perfect. Therefore, the next step of research would consider factors from more aspects, such as user tags, social relations and et.al., and further improve the accuracy.

Acknowledgements

This research is supported by the National Natural Science Foundation of China under grant 61271308 and the Fundamental Research Funds for the Central Universities, No. W17JB00060.

References

[1] G. Salton, M. McGill, An Introduction to Modern Information Retrieval, McGraw-Hill, New York, 1986.

[2] R. Mihalcea, P. Tarau, Textrank: bringing order into texts, in: Proc. 2004 Conference on Empirical Methods in Natural Language Processing, 2004.

[3] Z.-Y. Zhu, J.-J. Sun, Improved vocabulary semantic similarity calculation based on HowNet, Journal of Computer Applications 33(8)(2013) 2276-2279.

[4] R.-B. Wang, Z.-Q. Chen, J.-Z. Zhou, et al., Short texts semantic relevance computation method based on Wikipedia, Computer Applications and Software 32(1)(2015) 82-85.

[5] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proc. WWW 2013, 2013.

[6] J. Kim, J. Choi, B. Ko, L. Eunji, P. Kim, Extracting user interests on facebook, International Journal of Distributed Sensor Networks 2(2014) 1-5.

[7] M. Michelson, S. Macskassy, Discovering users' topics of interest on twitter: a first look, in: Proc. of the 4th Workshop on Analytics for Noisy Unstructured Text Data, 2010.

[8] K.H. Lim, A. Datta, Interest classification of twitter users using Wikipedia, in: Proc. the 9th International Symposium on Open Collaboration, 2013.

[9] J. Chen, R. Nairn, L. Nelson, M. Bernstein, Short and tweet: experiments on recommending content from information streams, in: Proc. International Conference on Human Factors in Computing Systems, 2010.

[10] F. Zhao, Y.-J. Zhu, H. Jin, L. T. Yang, A personalized hashtag recommendation approach using LDA-based topic model in microblog environment, Future Generation Computer Systems 65(2016) 196-206.

[11] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent dirichlet allocation, Journal of Machine Learning Research 3(2003) 993-1022.

[12] W. Wu, B. Zhang, M. Ostendorf, Automatic generation of personalized annotation tags for twitter users, in: Proc. Human Language Technologies: the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2010.

[13] A. Java, X. Song, T. Finin, B. Tseng, Why we twitter: an analysis of a microblogging community, in: Proc. Advances in Web Mining and Web Usage Analysis, 9th International Workshop on Knowledge Discovery on the Web, 2007.

[14] G. Cormode, M. Garofalakis, Sketching Probabilistic Data Streams, ACM Press, New York, 2007.