International Journal of Database Theory and Application
Vol.8, No.2 (2015), pp.107-118
http://dx.doi.org/10.14257/ijdta.2015.8.2.11

Content-Based Social Network User Interest Tag Extraction

Mei Yu[1], Xu Han[2], Xiaolu Gou[3], Jian Yu[4*], Fang Lv[5] and Jingyu Li[6]

Tianjin Key Laboratory of Cognitive Computing and Application,
School of Computer Science&Technology, Tianjin University, Tianjin, China
yumei@tju.edu.cn, hanxu2012@tju.edu.cn, gouxiaolu@tju.edu.cn,
yujian@tju.edu.cn,lvfang@tju.edu.cn, 1020365769@qq.com

Abstract

As a typical social network media, microblog has attracted a lot of users and quantities of information. More and more researchers and scholars are keen to explore useful information contained in microblog data and discover features of user interest for personalized recommendation. According to the unique feature of microblog text, this paper adjusts the traditional processes of Chinese word segmentation and tag extraction. Aiming to establish microblog user interest model, this paper combines clustering and classification algorithm to extract user interest tags. Experimenting upon Sina Weibo text dataset, the conclusion is reached that the method proposed in this paper is more effective and accurate to discover user features and the tags extracted are more in line with the user interest.

Keywords: *Social Network, Text Mining, Micro blog, Clustering*

1. Introduction

With the advancing of Internet, Social Networking Services, *i.e.* SNS, has developed rapidly. People understand SNS better than before and the use of SNS is more common [1]. Microblog, a typical SNS application, grows popularity rapidly among Internet users and gradually affects people's way of living and thinking [2]. In the microblog platform, users form personal relationships by following others. The information released is spread rapidly by the way of forwarding by users. These make microblog not only a social network service to realize social interaction, but also an important media to propagate information as well as comments [3]. However, the information in microblog is of great quantities. For users, how to get access to useful information, how to find like-minded friends in the large crowd of people, for advertisers, how to locate businesses advertising to gain maximum benefit, these all become hot research topics.

The messages the user released, forwarded or commented reflect one's hobby and interest. Extracting the user interest from these messages becomes the breakthrough in solving the problem raised above. Having studied and analyzed the microblog user data, the microblog user interest modeling is emphasized in this paper. Microblog data preprocessing and text vector clustering as well as classification are included.

While analyzing the microblog text data which has been removed stop-words according to generic stop-words list, it can be found that there still exist a lot of words of no value to extract user interest. Therefore, in this paper, removing stop-words which appear only in microblog data is added into data preprocessing step to reduce noise data for next research work.

Generally, in the process of modeling microblog user interest tags, the most important work is classifying the data vector to get tags. However, if the data vector is sparse, the classification results would be poor. In order to get more accurate tags, this paper considers to enhance the vector's density through clustering and then classify the clustering results. Experiment results show that the tags extracted in this way obviously conform to user features.

ISSN: 2005-4270 IJDTA

2. Related Work

There are quite a few scientific achievements about microblog user interest modeling. Huang He et al. used Formal Concept Analysis to establish user interest model from positive documents [4]. Goldberg *et al.* proposed "collaborative filtering" and developed recommendation system which named Tapestry [5]. In Paper [6], the research on Chinese Text Segmentation problem was conducted. Wealth et al. found the association between users and topics [7]. Yamaguchi et al. established microblogging-user relationship diagram according to the information relationship between users and micro loggings [8]. Adomavicius and Tuzhilin chose data mining techniques to mine the web surfing records of users and they combined the resulting association rules with personal registering information to get user personalization model [9]. Sofia Stamou and Alexandros Ntoylas obtained user interest model by analyzing the query words entered by the user and feed-back web theme information [10]. In Paper [11], Dino Isa et al. transformed document into a vector using Bayesian formula. Bansal et al. proposed a strategy to grab the latest spreading mciroblogging of the Twitter and then analyzed these data online to identify emergencies [12]. In Paper [13], the feature of the microblog in Twitter is summarized as short, of high noise, and with the trend of retweeting, then based on LDA model, a Twitter-LDA model was proposed to mine topics in twitter.

However, only the steps of removal of special punctuation such as "//", "\#\ ...\#" and "@", and removal of generic stop-words are contained in the preprocessing of microblog data by the scholars above. There is no strategy combining microblog users and data features to deal with words only appearing in microblog. In the extraction process of user interest tag, only clustering or classification is contained in the step, but the combining of them is not considered. In this paper, the method to model users in Chinese microblog application Weibo(short for Sina Weibo) was introduced after analyzing the two problems raised above. After successfully eliminating the interference of Weibo stop-words, meanwhile the user tags extracted are more in line with user feature. An algorithm of extracting Weibo user interest tag would be proposed in next section. Section 5 talks about the experiment and analysis. Finally, there is a conclusion about the advantages of the algorithm raised in this paper.

3. The Algorithm to Extract Weibo User Interest Tags

The messages a user published reflect one's interest. In this paper, the Weibo short text messages were used as research data. After the data was dealt with special stop-words of Weibo, the strategy combining clustering and classification was adopted to model users. Figure 1 is a model framework for Weibo user interest tag extraction algorithm.

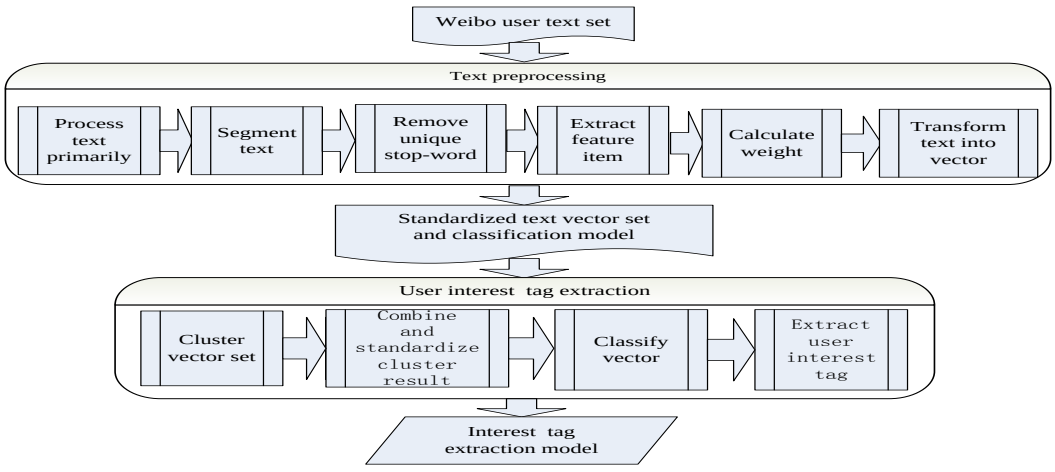

Figure 1. Model Framework for Weibo User Interest Tag Extraction Algorithm

In the User Interest Tag Extraction Algorithm, we proposed two sub-algorithms that are Weibo User Keywords Extraction Algorithm (WUKE for short) and Weibo User Tag Extraction Algorithm (WUTE for short).

3.1. WUKE Algorithm

WUKE algorithm focuses on the removal of stop-words appearing only in Weibo and its text vectorization space representation.

Weibo short text differs from other text, with no standardization. It is often informal, full of oral languages and daily life information, and also often includes widely used non-standard terms, noises and emoticons. These features add more difficulties to the understanding of these messages and event detection [14]. Therefore, these kinds of words need to be preprocessed. The WUKE algorithm starts with the step to remove time, numbers and special punctuation in the text. Then the remaining text should be segmented, and the stop-words would be removed according to the genetic stop-words list. After the stop-words being removed from the text, there are still words useless to model users. Considering that, the WUKE algorithm adds another step to process the data named depth pretreatment. In 1000 texts, 100 texts are randomly selected, and segmented, and then the term frequency value (explained below) of each item is calculated and sorted in descending order. This step should be repeated quite a few times. The items which always ranked high in the sorting results would be existed commonly in different texts. That kind of items would not contribute to modeling users. So in this paper, the top 15 items in each sorting result are selected as Weibo unique stop-words and should be removed from Weibo text.

Text content cannot be analyzed directly, therefore text vectorization is proposed. Vector Space Representation Model, put forward by Salton *et. al.* In 1980s [15], is a simple text vectorization algorithm. The dimension of a text vector is corresponding to the number of feature items in text itemset, while the value of a vector dimension is corresponding to the weight of feature item. The weight is calculated by TF-IDF (Term Frequency-Inverse Document Frequency) weighting method. TF-IDF is a method often used to calculate weight in quantifying text to vector. TF is term frequency, IDF indicates a percentage of the text in which feature item appears in whole texts. These two values can be calculated by Equation (1) and Equation (2).

$$TF_{ij} = \frac{n_{ij}}{\sum_{k} n_{kj}} \tag{1}$$

Where n_{ij} represents the number of appearances of feature item T_i in the text D_j.

$$IDF_i = \log \frac{|D|}{|\{j : T_i \in D_j\}|} \tag{2}$$

Where $|D|$ denotes the size of the text set D, and T_i represents the i-th feature item.

IDF values for the statistics of the prevalence of a feature item (keywords) in the entire text set. The smaller the IDF is, the more common the feature item is, and the less distinction or significance the term means. In this paper, joining the removal of stop-words that appear only in Weibo into the data preprocessing step, can effectively remove items which do not contribute to extracting interest feature. Therefore, the weight calculation method in this paper merely considers TF value of the feature item.

To reduce the error situations in text vectorization process, we need to normalize the text vector, as follows.

Let $\tau = (l_1, l_2, ..., l_{n-1}, l_n)$ be a vector. The value of each dimension is to be mapped into [0, 1], then the unitization normalization results is as Equation (3).

$$\psi = (\frac{l_1}{\sqrt{l_1^2 + l_2^2 + ... + l_{n-1}^2 + l_n^2}}, \frac{l_2}{\sqrt{l_1^2 + l_2^2 + ... + l_{n-1}^2 + l_n^2}}, ... \frac{l_{n-1}}{\sqrt{l_1^2 + l_2^2 + ... + l_{n-1}^2 + l_n^2}}, \frac{l_n}{\sqrt{l_1^2 + l_2^2 + ... + l_{n-1}^2 + l_n^2}}) \quad (3)$$

Where l_i denotes the value of the i-th dimension of Vector τ, ψ represents the normalization result of τ.

There are two reasons why using vector to represent text. One is that it is simple and clear, and the other is to prepare for clustering and classification in the next step of work. Only if its form is same with the one of training data vector, Weibo data vector can be classified by classification model. The dimension number of Weibo data vector matches the size of itemset formed by training corpus. Each dimension represents the same meaning with that of training data vector. Figure 2 is a flowchart of processing data. Table 1 is pseudocode implementation of training classification model.

Table 2 is the description of WUKE algorithm.

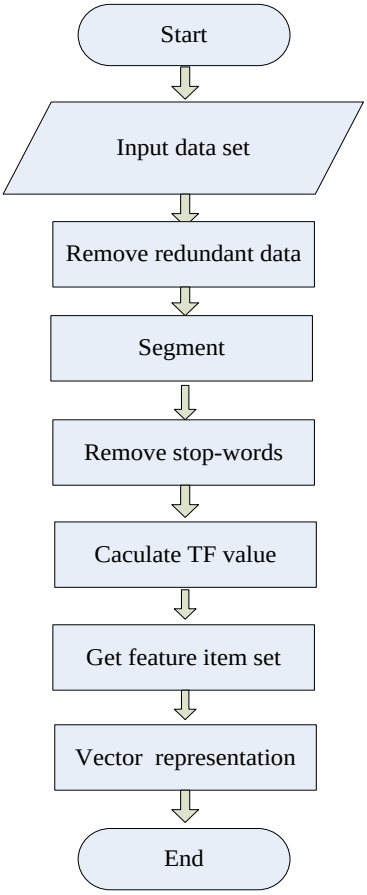

Figure 2. A Flowchart of Processing Data

Table 1. Pseudo Code Implementation of Training Classification Model

Algorithm 1: Train classification model
Input : training dataset S
Output : classification model M
1: $n \leftarrow$ the number of the texts in dataset S

2: for $i \leftarrow 0$ to n
3: create a text B_i to record the count of items in text t_i
4: $t_i \leftarrow$ remove time, number, #...#, @ and emoticons from t_i
5: $t_i \leftarrow$ Call Segmentation Model (t_i)
6: $j \leftarrow 0, k \leftarrow 0$
7: while (the item w_j is not the last item of t_i)
8: while(not the end of B_i)
9: if(w_j exist in B_i) then the corresponding count+1, $k \leftarrow 1$,break
10: if (k=0) then add w_j in the end of B_i ,and the corresponding count=0
11: $j++, k \leftarrow 0$
12: create a text C to record feature items
13: for $i \leftarrow 0$ to n
14: while(the item count in B_i >=30 and not the end of B_i)
15: write this item and corresponding count into C
16: for $i \leftarrow 0$ to n
17: the dimension number m of vector $\tau_i \leftarrow$ the size of C
18: $k \leftarrow 0$
19: while(k<=m)
20: the k-th dimension value $\leftarrow$ the count of k-th feature item in C
21: $k++$
22:
23: Call SVM($\alpha_1, \alpha_2, \cdots, \alpha_n$)
24: return M

Table 2. Description of WUKE Algorithm

Algorithm 2: WUKE algorithm

Input : Weibo text set S , feature item set C generated in training classification model, common stop-words list B
Output : standardized text vector set R
1: primarily preprocess the text set S , the result is recorded as S_1
2: use segmentation model to segment S_1 , the result is F
3: remove stop-words from F according to B ,the result is F_1
4: randomly repeat to select 100 texts in F_1 and sort items, choose the top 15 items in each sorting result to join into the Weibo unique stop-words list B_1, then preprocess F_1 depth according to B_1, record the result as F_2
5: then for each feature item in C , calculate TF value in F_2
6: generate text vector set: dimension number of a vector is the size of C , the value of i-th dimension is the TF value of i-th feature item
7: standardize the text vector set, gain the final text vector set
8: return R

3.2. WUTE Algorithm

WUTE algorithm focuses on combining clustering and classification in dealing with Weibo text vector, thus to get user interest tags.

Clustering algorithm needs to quantify the relationship between the vectors, i.e. the text similarity calculation. The most simple and popular method of this calculation is Manhattan distance, as shown in Equation (4).

$$d = \sum_{k=0}^{n} \left| x_k - y_k \right| \tag{4}$$

Where d is the value of similarity, $x_k (k = 0,1,..., n)$ is the k-th dimension value of the vector $x = (x_1, x_2,..., x_n)$, and $y_k (k = 0,1,..., n)$ is the k-th dimension value of the vector $y = (y_1, y_2,..., y_n)$.

After the text vector clustering, the vectors of each cluster would be combined. The step is that all vectors of each cluster would be summed to get one vector, then this new vector would be normalized to obtain vector representative of this cluster. Finally these representative vectors would be classified to get interest tags.

In WUTE algorithm, the Weibo text vector set produced by WUKE algorithm is used as the input. It would be clustered with cluster number being set as k. The k clusters obtained should be processed to get k representative vectors which are the input of classification algorithm. After the classification, the cluster tag to which each representative vector belongs would be obtained ultimately. These cluster tags are Weibo user interest tags. Figure 3 is a flowchart of WUTE algorithm.

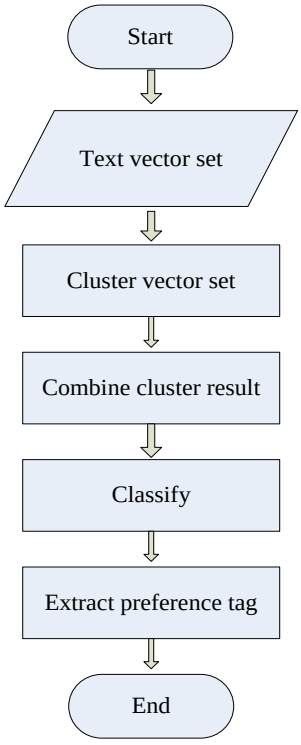

Figure 3. WUTE Algorithm Flowchart

4. Experiment and Analysis

4.1. Weibo Dataset

The experimental dataset is based on information released by Sina Weibo user. The Weibo user with the ID of 1773668752 is selected as a central node. The 127 Weibo users related with central user are treated as data source. All the data used for experiment are collected from Sina Weibo.

4.2. Training Dataset for Classification Model

Classifying text vector needs to be based on classification model formed by a lot of trained data. In this experiment, the Sogou Chinese dataset is chosen as training corpus for training model.

4.3. Segmentation Module

ICTCLAS from CAS is an HMM-based segment model. It can achieve great segmentation effect. In this study, the ICTCLAS method is used to preliminary preprocess Weibo text data.

4.4. Classification model

In this experiment, Sogou corpus is used as training dataset. The statistical topic tags of this corpus are shown in Figure 4.

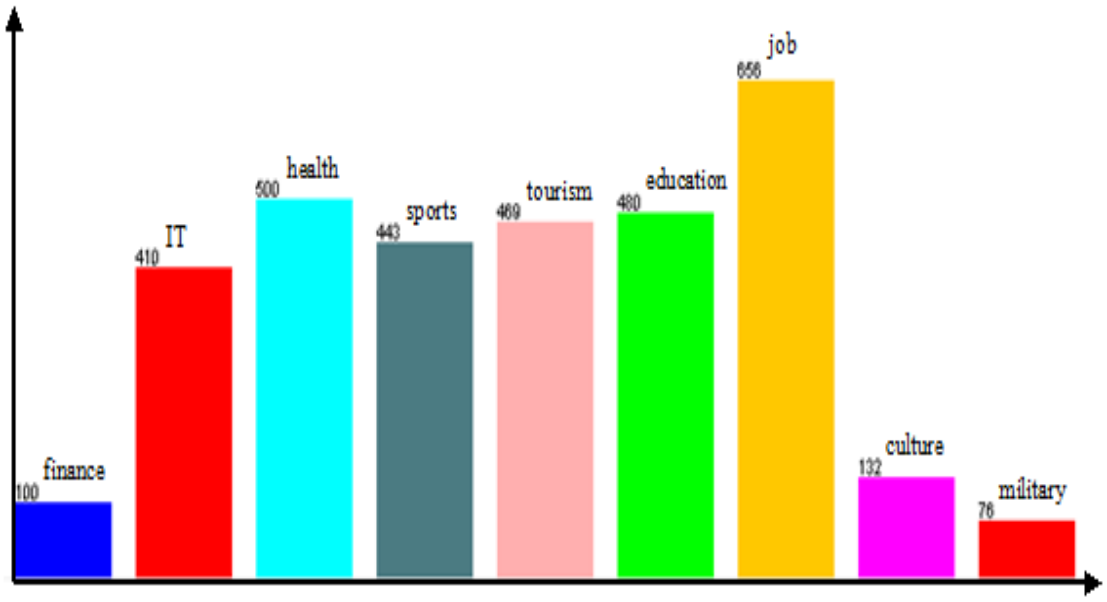

Figure 4. A Histogram of Topic Tags of Training Corpus

Vector result is the input of SVM algorithm to establish classification model. Figure 5 is classification model trained through Sogou Chinese corpus, wherein the horizontal axis is the text category coding while ordinate indicates text vector. Each point represents a text vector and class attribute of this text vector.

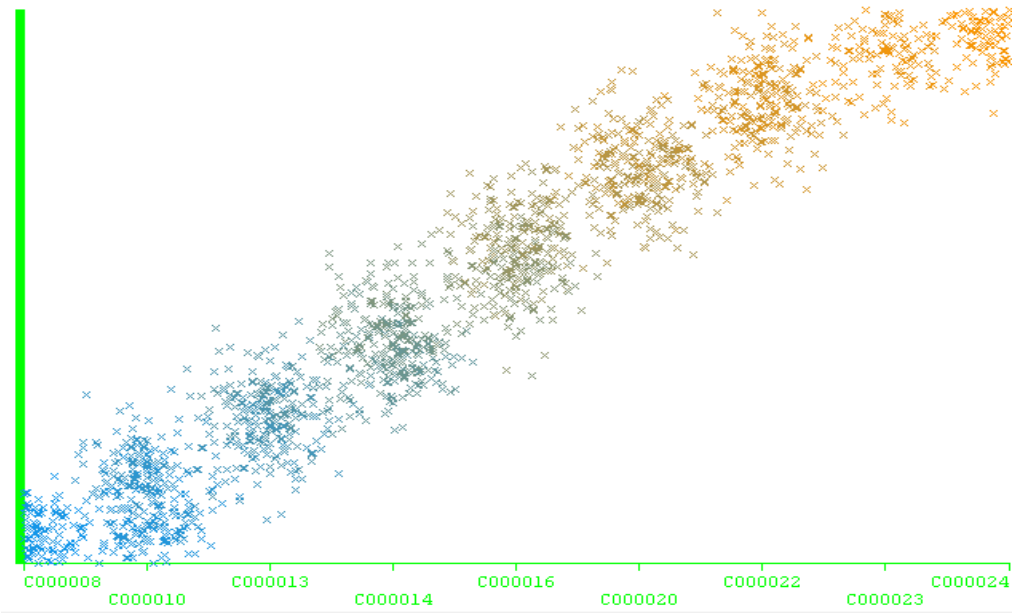

Figure 5. SVM Classification Model Trained through Sogou Chinese Corpus

4.5. Clustering on Weibo Data

The similarity between text vectors is measured using Manhattan distance. The PAM clustering algorithm, usually used for small-scale text set, is to cluster the Weibo text vector. The number of clusters is 6.

4.6. The Comparison between HMM and WUKE

The Weibo text released by User 1778383043 is considered as experiment data. The HMM is used as baseline method. The method based on HMM model and the WUKE algorithm raised by this paper are used separately to process the experimental data to obtain the user interest feature items. The results are shown as Figure 6 and Figure 7. The main feature keywords extracted from Figure 6 and Figure 7 are shown in Table 3.

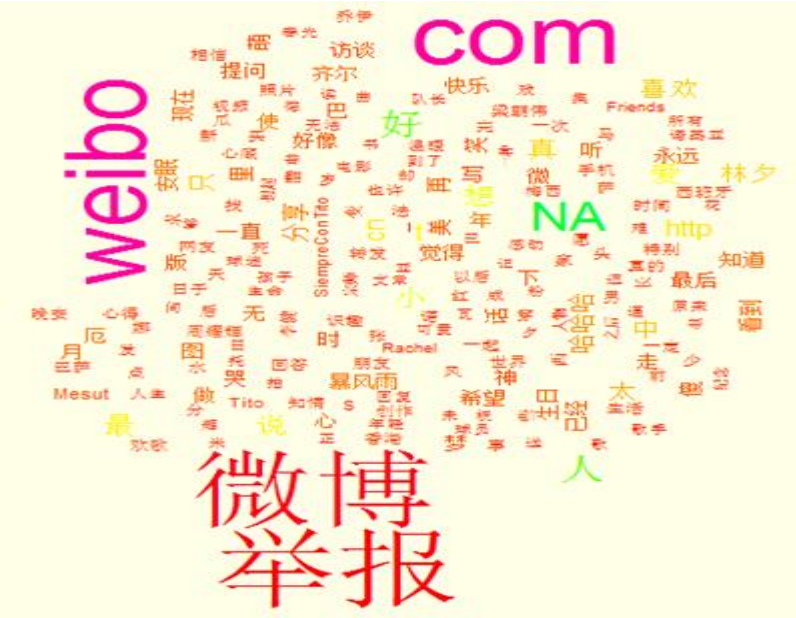

Figure 6. The Experiment Result of HMM Algorithm

Figure 7. The Experiment Result of WUKE Algorithm

Table 3. Main Keywords Results from Different Two Data Preprocessing

Preprocessing method	Weibo feature keywords						
HMM algorithm(only)	微博	举报	weibo	com	NA	好	好像
WUKE algorithm	林夕	喜欢	哈哈哈	分享	快乐	太	周耀辉

In Table 3, feature keywords extracted by HMM algorithm include “微博”, “举报”, “weibo”, “com”, “NA” and so on. They are all not emotional, and very common. The user's taste and features can hardly be analyzed from these items, so they are of no value to analyze user interest. Keywords extracted by WUKE algorithm contain “林夕” and “周耀辉”. These two items imply that the user likes the songs written by “林夕” and “周耀辉”. From that it can be known that the user is fond of music and appreciates creators. “喜欢”, “哈哈哈” and “快乐” show that the user is interested in jokes. It can be seen that compared with traditional HMM algorithm, the WUKE algorithm can be more helpful in finding user features. Therefore, the WUKE algorithm introduced by this paper can make great achievement in analyzing user interest.

4.7. The Comparison between SVM and WUTE

The result of WUKE is considered as experiment data. The WUTE algorithm is used to process experimental data. The number of clusters is set as 6. The clustering result is shown in Figure 8. Different clusters are represented by different colors. Elements in the same cluster share the same color. Each cluster is surrounded by oval.

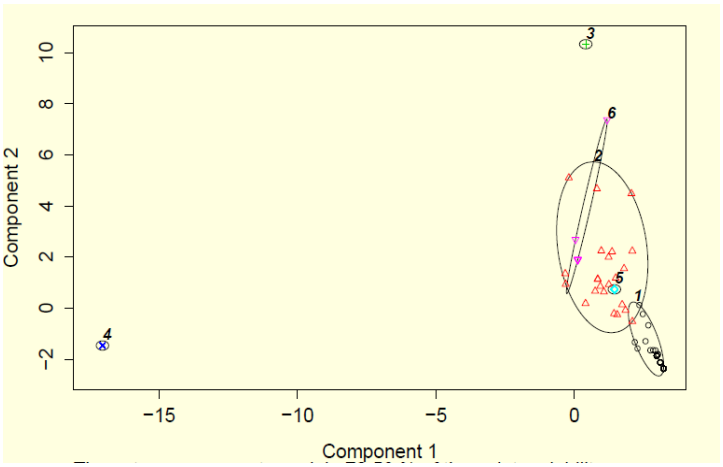

Figure 8. The Clustering Result of User 1778383043 Data and k is Equal to 6

It can be seen from the clustering result, there are totally 188 Weibo texts released by the user. All the texts are divided into six sets with the number of 103, 23,18,22,14 and 8, separately belong to 6 clusters. After the clustering result being combined, they are applied in the classification model of WUTE algorithm. The classification result is shown as Figure 9. In Figure 9, the tags the 6 vectors belong to can be seen clearly. Thus the user tags are tourism (C000016), culture (C000023), and job (C000022).

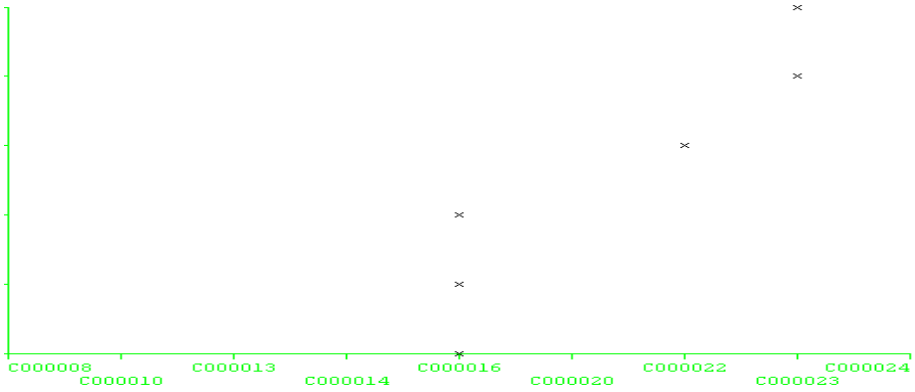

Figure 9. Weibo Text Classification Result (WUKE algorithm)

The result of using SVM to handle the experiment is shown in Figure 10.

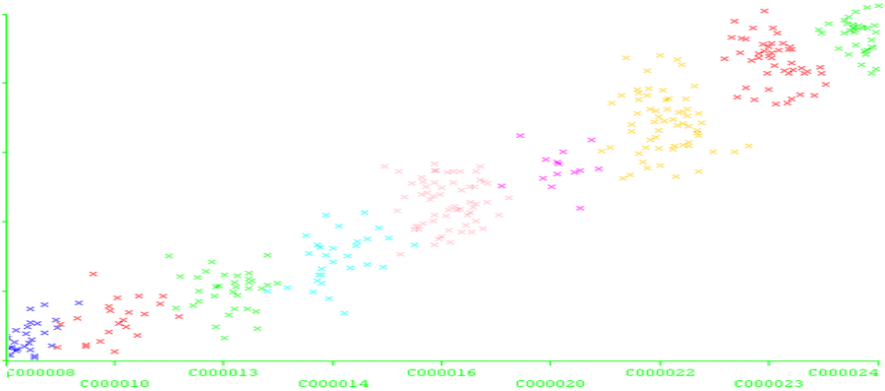

Figure 10. Weibo Text Classification Result (SVM algorithm)

Obviously, the Weibo text vectors are distributed by SVM algorithm into different clusters. To get user interest tags, the threshold is set as 21 (averaged), that is if the number of the Weibo text vectors in cluster is more than 21, the tag of this cluster would be added into the user interest tag set. As shown in the results, user interest tag set contains health (C000013), sports (C000014), tourism (C000016), job (C000022), culture (C000023).

From the comparative analysis of Figure 9 and Figure 10, it can be seen that WUTE algorithm, combining clustering and classification, is more intuitive, representative than SVM algorithm. In addition, in the processing of SVM algorithm, in order to define user tags, the threshold to determine whether it is user interest tag has to be set manually. However, to choose the threshold is subjective. WUTE algorithm could avoid this defect.By using the WUKE and WUTE, the Weibo user interest tags extraction process can avoid noise data and the sparse problem of the text vectors, thus to increase the effectiveness of the classification results to obtain useful tags.

5. Conclusion

Unlike other texts, microblog texts contain some words that only appear in microblog, besides special punctuation and emoticons. In this paper, the microblog user interest tags extraction algorithm is designed considering the features of microblog texts. On one hand, the step of removal of microblog unique stop-words in data pretreatment process could filter out the items with no sense for extracting user interest tag thus to shorten the time of selecting tags. On the other hand, extracting tags by the method that combines clustering and classification could improve the effect of classification and improve the accuracy of user interest tags extracted. By validating on Sina Weibo user dataset, it is confirmed that the designed algorithm could reduce the interference of noise data in microblog text preprocessing process. Also, it can effectively solve the sparse problem of microblog text data in tag extraction process, thus the designed algorithm is more helpful to model user interest.

Acknowledgements

The author would like to thank the anonymous reviewers for their helpful and constructive comments.

References

[1] H. Cheng, Y. Liu, J. Li, J. Zhu, and J. J. Cheng, “Content-based Micro Blog User Preference Analysis”, EN, vol. 7, no. 1, **(2012)**, pp. 282-289.
[2] M. Sahami, and T. D. Heilman, “A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets”, Proceedings of the 15th International Conference on World Wide Web, Edinburgh, UK, **(2006)**, pp. 377-386.
[3] H. Lieberman, “An Agent that Assists Web Browsing”, Proceedings of the International Joint Conference on Artificial Intelligence, Montreal, **(1995)**, pp. 924-929.
[4] H. Huang, H. Huang, and R. J. Wang, “FCA-Based Web User Profile Ming for Topics of Interest”, Proceedings of the 2007 IEEE International Conference on Integration Technology, Shenzhen, China, **(2007)**, pp. 20-24.
[5] D. Goldberg, D. A. Nichols, B. M. Oki, and D. B. Terry, “Using collaborative filtering to weave an information tapestry”, Communications of the ACM, vol. 35, no. 12, **(1992)**, pp. 61-70.
[6] T. Brants, F. Chen and I. Tsochantaridis, “Topic-based document segmentation with probabilistic latent semantic analysis”, Proceedings of the 11th International Conference on Information and Knowledge Management, **(2002)**, pp. 211-218.
[7] M. J. Welch, U. Schonfeld, D. He, and J. Cho, “Topical semantics of twitter links”, Proceedings of the fourth ACM international conference on Web search and data mining, New York, USA, **(2011)**, pp.327-336.
[8] Y. Yamaguchi, T. Takahashi, T. Amagasa, and H. Kitagawa, “TURank:twitter user ranking based on user-tweet graph analysis”, Proceedings of the 11th international conference on web information systems engineering, Berlin, **(2010)**, pp. 240-253.

[9] G. Adomavicius, and A. Tuzhilin, "Using Data Mining Methods to Build Customer Profiles", IEEE Computer, vol. 34, no. 2, **(2001)**, pp. 74-82.
[10] S. Stamou, and A. Ntoulas, "Search personalization through query and page topical analysis", User Modeling and User-adapted Interaction, vol. 19, no. 1-2, **(2009)**, pp. 5-33.
[11] D. Isa, L. H. Lee, V. P. Kallimani, and R. Rajkumar, "Text Document Pre-Processing Using the Bayes Formula for Classification Based on the Vector Space Model", Computer and Information Science, **(2008)**.
[12] N. Bansal, and N. Koudas, "A System for Online Analysis of High Volume Text Streams", In Very Large Data Bases, **(2007)**, pp.1410-1413.
[13] W. X. Zhao, J. Jiang, J. S. Weng, J. He, E-P. Lim, H. F. Yan, and X. M. Li, "Comparing twitter and traditional media using topic models", Proceedings of the 33rd European conference on Advances in information retrieval, **(2011)**, pp. 338-349.
[14] Z. T. Liu, W. C. Yu, W. Chen, S. Wang, and F. Y. Wu, "Short Text Feature Selection for Micro-Blog Mining", Proceedings of International Conference on Computational Intelligence and Software Engineering, Wuhan, China, **(2010)**, pp. 1-4.
[15] G. Salton, M. Mcgill, "Introduction to modern information retrieval", In Computer linguistic, (1984).

Authors

Mei Yu, is an associate professor in School of Computer Science & Technology, Tianjin University. Her research interests are in peer-to-peer network, wireless network, data mining and database application technology.

Xu Han, is a graduate student at the School of Computer Science & Technology, Tianjin University. His research interests include Data Mining and sentiment analysis.

Xiaolu Gou, is a graduate student at the School of Computer Science & Technology, Tianjin University. Her research interests include Data Mining and commendation algorithm.

Jian Yu, is now an engineer in School of Computer Science & Technology, Tianjin University. His research interests are in wireless network, data processing, and database application technology.

