International Journal of Hybrid Information Technology
Vol. 10, No. 9 (2017), pp.29-40
http//dx.doi.org/10.14257/ijhit.2017.10.9.03

Popular Microblogging Quality Prediction Model using Potential Fusion Characteristics

Shaowei Li[1] and Chengying Chi [2,*]

[1,2,]School of Software Engineering, University of Science and Technology ,Liaoning, Anshan 114051, China*
[1]lixuanming526@163.com, [2,]chichengying@ustl.edu.cn*

Abstract

In this paper, we put forward a potential fusion feature model to deal with the popularity prediction problem for Sina Microblogging. We analyze the characteristics of microblog forwarding number, number of views and comments. By combining the implicit relation between features, we fuse the n-gram model to establish a quality prediction model for microblogging. The characteristics of the model are combined with the regression model to predict the popularity of a microblog text. Our experiment results show that the model is effective.

***Keywords**: text quality prediction; potential fusion characteristics; n-gram model; logistic regression; popular microblogging*

1. Introduction

In recent years, online social networking [1-3] is developing rapidly. Microblogging is one of the typical representatives. Microblogging is a broadcast medium that exists in the form of blogging. A microblog differs from a traditional blog in that its content is typically smaller in both actual and aggregated file size. Microblogs allow users to exchange small elements of content such as short sentences in a timely manner. As a new carrier of information dissemination, microblogging has been widely used all over the world[4]. Among the mainstream microblogging service sites, undoubtedly, Twitter is the most popular in the western world, while in China, Sina Microblogging is the most popular (user number: 341 million as of Dec. 2016). However, in the current era of information explosion, there always is a mass of information being generated and updated. How to allow users receive high quality and interesting information within a limited time, so as to improve their experience is a primary task. In response to such a demand, not long ago, Sina Microblogging launched a new "hot microblogging" section and set up a special department for business development to deal with such requirements, but the user experience is not as good as expected, so further efforts to improve the system is necessary. To solve this problem, we have established an automatic prediction model which effectively improved the accuracy of popular blog prediction and user experience.

Although there is no research literature devoted directly to popular microblogging, the methods in the related literature are helpful to the construction of the model, such as microblogging feature analysis, emotional tendencies [5], and forwarding prediction.

Total number of comments [6,7], total forwarding number [8], and total number of views [9,10] form the most common microblog popularity characteristics. It is worth noting that a single characteristic as an index of popularity is justified, but it is too one-sided, and can not reasonably evaluate the quality attributes of the text, nor can it provide accurate quality prediction for microblog texts. Therefore, the fusion feature standard

Received (April 19, 2017), Review Result (August 17, 2017), Accepted (August 23, 2017)
[*] Corresponding Author

came into being, that is, multi-feature fusion [11]. Specific multi-feature fusion does not have absolute criteria. Specific characteristics for different areas need to be analyzed.

Microblog popularity prediction method is mainly based on microblog text forwarding prediction [12]. For the microblog text forwarding prediction there are several methods which are based on the user's past behavior, the user's text interest, the user's influence on the group, or the mixed feature learning. Of all these methods, the one based on mixed feature learning is the most simple and intuitive, and the explanation of the model is weak, but it depends on the selection and combination of features.

This paper describes the potential fusion feature model constructed by fusing the interaction exposure feature and the potential connection between the forwarding number and the number of views into the N-gram model. The project is a cooperation with Sina Microblogging, and all the source data in this paper are derived from the Sina Microblogging official database, and the data sets are preprocessed by sampling. In the model training, through the theoretical analysis and experimental comparison, we finally adopt a regression model that can better match the data characteristics, that is, Spark Machine Learning (Spark ML) in the Logistic regression (LR) model for potential fusion characteristics model training.

2. Feature-based Data Acquisition and Preprocessing

For the microblog text feature, the selected data set refers to the ID number, text field, text interaction number (total forwarding number) of the microblog texts, the number of text impressions (total number of views), and the textual content of the texts. The interactive number of text and the number of text exposure are used to obtain the interactive exposure ratio (Pro).

2.1. Selection of Data Sets

According to the microblog text feature and the selected data set, the above mentioned text feature is extracted from the source data and processed according to the required conditions and requirements.

Text data extraction and processing is carried out through microblogging official Hadoop cluster under the Hive data warehouse module to filter the data. From the source database, we selected the number of interactions (act_num) >= 10, the number of impressions (recommended_num) >= 50, and the microblogging issued for the previous day’s last activity record (that is, the number of interactions and exposures are the largest) of all the microblogging corresponding to the above information, and calculate the potential feature - interactive exposure ratio (Pro), the formula is as follows.

$$\mathrm{Pro} = \frac{\mathrm{act\_num}}{\mathrm{propose\_num}}\% \tag{1}$$

We put the Hive filtered data table into different files according to their sub-domain to wait for the next step.

In order to eliminate the relevant noise caused by the relevant information, after the above processing, we add another step to delete the overlapping text ID to repeat and overlapping text contents, and then do the critical noise boundary processing. For the value of the interactive exposure ratio, we take the upper boundary value and discard the lower boundary value.

2.2. Text Preprocessing

The preprocessing of microblog text is roughly the same as conventional text preprocessing. However, we have made special treatment for microblog text in some technical aspects, and constructed a microblog short text preprocessing system. The system involves the technology and process shown in Figure 1.

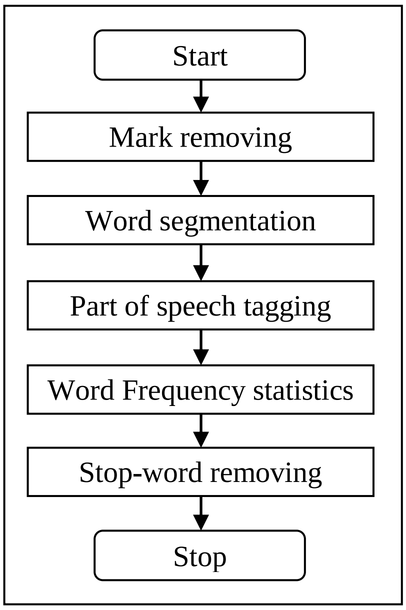


Figure 1. Text Preprocessing Module System Diagram

2.2.1. Mark Removing

The main purpose is to remove some text content that is irrelevant or contributes little to the textual characteristics. In the microblog short text, such as short chain (such as URL), appears frequently, but for the text feature extraction, it is not helpful, and should be removed.

Short chain examples in our Microblogging database are as follows: http: t.cn/RZg8G4Y; http: t.cn/R5RkLhQ;http://t.cn/R5d6anT and so on. Microblogging uses its unique way to edit the short chain, so we use regular expression to represent short chain for processing.

2.2.2. Chinese Word Segmentation

Chinese word segmentation and part of speech tagging are both the initial stages of natural language processing and one of the basic contents of Chinese information processing. Since the microblog short text is no longer than 140 words, most of the word segmentation tools can meet the word segmentation needs. But the content of microblogging has its own unique characteristics, therefore, in the comprehensive consideration of text features, time and space efficiency and other factors, we selected a freeware toolkit "jieba" for word segmentation.

2.2.3. Part-of-Speech Tagging

Part-of-speech tagging is mainly for the removal of punctuation and other irrelevant words. However, special symbols in Chinese texts require further treatment, or a second word segmentation may be necessary for long words which may initiate a recursive process.

2.2.4. Word Frequency Statistics

Word frequency count is a vocabulary analysis method. By counting the number of occurrences of each word in a certain length of the corpus, the statistical results may be used in subsequent processes.

2.2.5. Stop Words Removing

Stop words are words which are filtered out before or after processing of natural language texts. They are typically the most common words with high frequency, such as such as "the", "and" in English. In Chinese, they usually are auxiliary words.

In our system, we use the word frequency statistics to get dynamic distribution of the vocabulary, and generate a stop word list to remove the corresponding stop word.

At this point, the text preprocessing system has been basically completed.

3. Feature Extraction

The contextual semantic relevance of text has always been a very important issue in the field of text processing. For Chinese microblog text there is no exception. The main algorithms for dealing with context semantic associations are N-shortest path algorithm [13], N-maximum probability algorithm [14], maximum matching algorithm [15], n-gram model, and perceptual algorithm. Due to the characteristics of the microblog short text, the selected features are necessary to ensure the independence, that is, they have a relatively high independent representation and context free. But we also have to consider the relationship between features, that is, context semantic association. Comparing the above methods, we conclude that the N-shortest path algorithm satisfy the binary relation, and the context semantics association degree is not high. The N-maximum probability algorithm is more suitable for long text, and it is relatively suitable for microblogging short text. Although the maximum matching algorithm is suitable for microblog text, its matching length is not easy to control, for short text more need for relatively stable algorithm; perceptual algorithm is more suitable for binary word matching, this does not apply. In contrast, the n-gram model is more advantageous, and it is suitable for the length of text, matching length control, and the use of hidden Markov model, which can meet the characteristics of their own independent, while ensuring the control between features association. In this project, the popular microblogging application is based on the text feature analysis of the user's emotional preferences for emotional classification and recommendation, literature [16] has done much of related work, and theory and practice has proved the n-gram advantage. Therefore, we choose the n-gram model.

3.1. Creating an N-Gram Model

Based on the characteristics of the microblog text, this paper chooses the n-gram model as the basic feature model. Here we choose the 1-gram, 2-gram, 3-gram gram model. Among them, 1-gram solves the problem of independence of features, 3-gram solves the problem of text context semantic association, and 2-gram is a smooth transition of the above two feature models. As the microblog text is short text, rather than long text, therefore, the above three kinds of gram can achieve a relatively comprehensive effect. Too many features will bring about the trouble of processing, result in data characteristics of the over-fitting, increase the complexity of the model, and reduce the generalization of the model. And fewer characteristics can cause data features less fit, making Model prediction accuracy too low. The above problems will affect the actual effect of the model. Therefore, we use the above three kinds of n-gram model as the basic carrier of the feature fusion model, in order to achieve the global optimization of the model.

3.1.1. Text Word Segmentation and N-Gram Initial Structure

After processing the text as described in 2.2, we can get 1-gram and its related information, and with different stitching and matching we can be 2-gram, 3-gram and their associated information. After the information processing such as repeatability check and content similarity check, the resulting n-grams are expressed as follows.

1-gram:

$$T(d_i)=(w_1(d_1),w_2(d_2),...,w_m(d_n)) \tag{2}$$

2-gram:

$$T_2(d_i)=(w_1(d_1d_2),w_2(d_2d_3),...,w_m(d_{n-1}d_n)) \tag{3}$$

3-gram:

$$T(d_i)=(w_1(d_1d_2d_3),w_2(d_2d_3d_4),...,w_m(d_{n-2}d_{n-1}d_n)) \tag{4}$$

where n is the number of features of the dimension of word vector obtained after preprocessing, w_i is the ith feature of n-gram, and d_i is the ith characteristic of the word vector after preprocessing.

3.1.2. Handling N-Gram with Potential Links According to The Required Characteristics

By counting the number of characters that appear in each of the elements in the n-gram, retaining the corresponding text number, and removing the duplicate n-gram elements, we get the following.

$$V_{mid}=(gram_i, mid_1, mid_2,...,mid_i,...,mid_n), i\in(1,n) \tag{5}$$

where $gram_i$ is the element in the n-gram, mid_i is the ID of the text corresponding to the gram, and n is the total number of text that appears in $gram_i$.

3.1.3. Extracting N-Gram Features

We then perform a distribution statistics of text frequencies in the n-grams, and accordingly, set up the threshold. The corresponding formulas are as shown below:

$$\begin{cases} Tv_{low}=\alpha*\sum_{i=1}^{n}w_i, \alpha\in[0.11,0.18] \\ Tv_{up}=\beta*\sum_{i=1}^{n}w_i, \beta\in[0.59,0.72] \end{cases} \tag{6}$$

where n is the total number of elements for each n-gram, and w_i is the corresponding element, all is quantized to 1; α is the lower threshold coefficient, and β is the upper threshold coefficient. α and β are different from different areas of text, different numbers of text, according to a large number of different data after the experiment, the value of α is stable between 0.11 and 0.18, and the value of β is stable between 0.59 and 0.72. The approximate optimal solution of the experiment appears when the threshold coefficients are set to α = 0.15 and β = 0.67. Therefore, we use these values to extract the threshold for the n-gram feature.

3.1.4. N-Gram Fusion Interactive Exposure Ratio Distribution Characteristics

For the results in 3.1.3, the degree of concentration (Con) (result preserves the three significant digits) is calculated for the interactive exposure ratio (Pro) corresponding to the text ID number (mid) included in each n-gram element. Set the threshold (Tv) according to the statistical distribution result, and select the n-gram with the degree of concentration less than the threshold.

The degree of concentration is calculated as follows:

$$Con = \frac{\mathrm{Pro}_{\max} - \mathrm{Pro}_{\min}}{\sum_{i=1}^{n} i} \tag{7}$$

The threshold setting varies depending on the text data in different areas and the amount of different text data, but the principle is the same. They are in accordance with the above method, the statistical distribution, the inflection point is the threshold, which is also an approximately optimal solution.

3.2. Building VSM

Vector space model (VSM) [17] was proposed by G. Salton of Cornell University. The VSM of this paper is also based on his idea. But in feature extraction, we use a combination of 1-gram, 2-gram, and 3-gram.

3.2.1. Build N-Gram VSM

The 1-gram, 2-gram, and 3-gram of 3.1 are combined to form a high-dimensional vector space model. Experiments show that this is more representative than a single n-gram. At the same time, after combining Pro, the combination of features is more representative, and the prediction accuracy is improved.

The VSM model is as follows:

$$V = \left(w_1\left(fea_1\right), w_1\left(fea_2\right), \ldots, w_1\left(fea_i\right), w_2\left(fea_1\right), w_2\left(fea_2\right), \ldots, w_2\left(fea_j\right), w_3\left(fea_1\right), w_3\left(fea_2\right), \ldots, w_3\left(fea_k\right)\right) \tag{8}$$

where i, j, and k represent 1-gram, 2-gram, and 3-gram of the respective dimensions, fea_i, fea_j, and fea_k are 1-gram, 2-gram, and 3-gram feature elements, respectively. $w_1\left(fea_i\right)$, $w_2\left(fea_j\right)$, and $w_3\left(fea_k\right)$ represent respectively, the characteristic words corresponding to 1-gram, 2-gram, and 3-gram, and their associated text weights. In *V*, this is expressed as equivalent to fea, in the associated text of its weight for additional storage, which will be used in later operations.

3.2.2. Constructing Text VSM Sparse Matrix

We use the sparse input format module libSVM in Spark to construct the texts' VSM matrix. And we use the elements in V to match the text content for the text space. If it matches, the vector space model (V_T) is added in the format of (subscript: weight).

The VSM for a single text is expressed as follows:

$$V_T = \left(\mathrm{Pro}, flag_1 : w\left(flag_1\right), flag_2 : w\left(flag_2\right), \ldots, flag_n : w\left(flag_n\right)\right) \tag{9}$$

The VSM corresponding to all the text is as follows:

$$V_{all} = (V_{T1}, V_{T2}, V_{T3}, \ldots, V_{Tn})^T \tag{10}$$

where Pro is the interactive exposure ratio of the text, and flag$_i$ ($0 \leq i \leq n$) is the dimension subscript corresponding to the feature in V in the text, and w (flag$_i$) is the number of occurrences of the feature in the text. In conventional VSM, only 0 and 1 are used as a feature of whether a match to the characteristics occurs. Although this can achieve a certain effect, the representation is not strong. And because the microblog texts are short text, and does not involve multiple documents (that is, without considering the problem of similarity between different domains), there is no need to consider the traditional TF * IDF representation. Here is just the record weight without too much complex calculation. We can meet the microblogging short text related needs. Therefore, we choose this relatively simplified TF * IDF method.

4. Model Training and Analysis

In this paper, the interactive exposure feature (pro) is a continuous function on the interval of 0-1. The traditional two classification problem usually chooses the intermediate value as the classification threshold and is set by the unit-step function. If the monotonic can be met, the alternative function can be chosen to improve, usually logistic function is chosen as an alternative. Literature [18] used this function in forecast model for in-depth analysis, and summed up the model characteristics. In fact, the processed microblog texts are almost completely in line with the characteristics of this function, which laid the theoretical basis of the transfer function of the training model used in this paper.

In practice, we draw on the conclusions in [19], and then use the Logistic regression (LR) algorithm in Machine Learning (ML) to train the model.

The model training algorithm is based on the LR model interface under Spark, combined with the real data set, using the libSVM data input format processing module to enter the data format processing and reception, and the data go through the libSVM interface into the LR model for training and test.

4.1. Building Positive and Negative Examples

In V$_{all}$, we sort the V$_{Ti}$ in descending order by Pro. And extract the top 35% of the data set as positive examples, and extracting the bottom 35% of the data set as negative examples. The rest of the data set is used as the validation data set. The positive set of labels is 1 and the negative example set is 0. The above-mentioned processed positive and negative examples are used as the data set of the model.

The positive and negative data set model is represented as follows:

$$V_D = \begin{cases} (1,\ flag_1 : w(flag_1),\ flag_2 : w(flag_2), \ldots, flag_n : w(flag_n)),\ positive-set, V_{D+} \\ \\ (0,\ flag_1 : w(flag_1),\ flag_2 : w(flag_2), \ldots, flag_n : w(flag_n)), negative-set, V_{D-} \end{cases} \tag{11}$$

4.2. Building Training Set and Test Set

In the machine learning phase, the processed data is generally divided into three types, training set, test set, and validation data set. In our case, only the definition of the characteristic parameters is involved in the model, but not the definition of the superparameters, and the relative parameters involved in the multiple residuals, fitting optimization and so on, and the expected results are similar to the two categories. Therefore, we only use the training set and the test set as the model data source.

For the training set and the test set, the data set of this paper is constructed as follows. The positive and negative examples were evenly distributed and smoothed, and 80% of the data were taken as the training set and 20% of the data was used as the test set to ensure high smooth and low data coupling.

4.3. Model Training

The model training and testing in our experiments is based on the Hadoop cluster environment, and the model training program is written by applying the Logistic regression (LR) model interface in the Spark machine learning component. According to the work on the source data and the work on the data set, adjustments have made for the data sets to suit the parameters of the model interface and the data input format. Using the training sets and test sets, we train and test our feature models. The model is trained and tested by different amount of source data. The results are tested by regular cross tests and the results of each comparison are recorded and compared to obtain the approximate optimal model.

5. Experiment and Analysis

In literature [20], natural language processing technology is used to analyze the contents of the microblog text, such as emotional tendencies and naming entities. It is found that the user characteristics and content characteristics can be complementary, and the combination can improve the accuracy of the forecast. In [21], the influence of the content of the microblog text on the forwarding amount is analyzed from the perspective of language expression, which proves the influence of different wording and language habit on microblogging forwarding. This experiment combines the ideas of the above documents and combines their respective advantages. Through experiments, the new method of thinking is applied to the prediction of microblog text quality. Our experiment results have shown that our method is effective, the following is a comparison of the results and analysis.

5.1. Comparison of Different Feature Models

The contrast experiment of the characteristic model is mainly used in the following three characteristics of the fusion model for exploratory comparison experiments, comment interact ratio n-gram, comment exposure ratio n-gram, and interactive exposure ratio n-gram. The following figure shows the results of these comparison experiments.

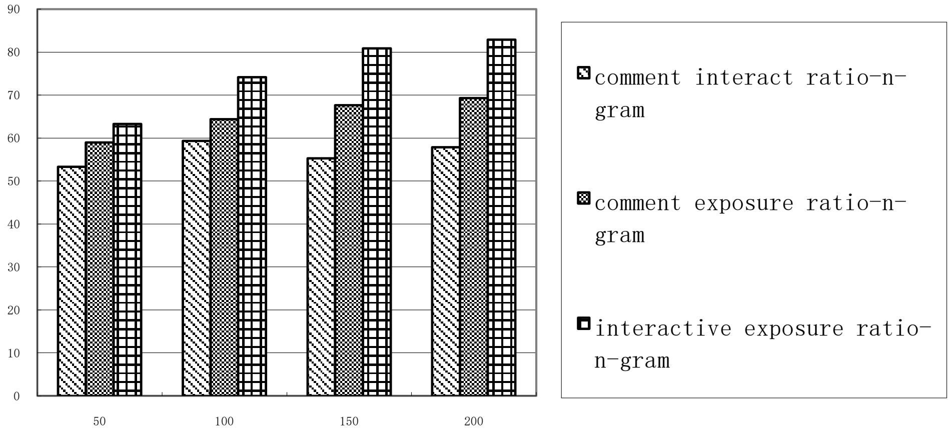


Figure 2. Comparison of Accuracy Scores for Different Feature Selections

The three different histograms in Figure 2 represent the scores of the three different combinations of features after the n-gram fusion. The horizontal axis represents the source data (100 million) and the vertical axis represents the accuracy score (%). As can be seen from Figure 2, regardless of the amount of source data, interactive exposure ratio n-gram potential fusion characteristics of the text quality prediction accuracy, shows an absolute advantage. The results also shows that comment interaction ratio n-gram is not representative, the model does not have to solve the practical problems of the role and value. As for the comment exposure ratio n-gram, although it has a certain practical significance, in response to the quality of the text rather than for the user's case, and it cannot provide a higher accuracy prediction role. Therefore, through the comparative experiment analysis, we choose interactive exposure ratio n-gram as a potential fusion feature model. In this paper, the prediction accuracy of the model is improved with the increase of the amount of text. However, when the number of texts reaches a certain amount, the prediction accuracy of the model tends to be stable and the accuracy is satisfying.

5.2. Model Comparison for Different Source Data Volumes

Table 1. Model Test Results

Source data（billion）	Processed data	Training sets	Test sets	Model dimension	Accuracy（%）
50	87594	18149	4537	40879	63.243
100	169696	50229	12557	77859	74.176
150	241885	71598	17900	113738	80.834
200	327895	97056	24264	138114	82.901

As can be seen from Table 1, for the model proposed in this paper, with the increase of the amount of source data, the number of training sets and test sets increases, and the three data sets are always linear. This also explains the correctness of the theory of this article, as well as the consistency of the experiment, which is consistent with the idea of linear processing experience. However, the dimension of the model is not the same. When the amount of data increases to a certain extent, the increase of the model dimension is gradually gentle and stabilized. At the same time, the accuracy of the forecast is basically stable above 80%. This further proves that the accuracy of the model depends on the model's data feature mining and multiple fusion. The model proposed in this paper is derived from this premise, and obtained through experimental comparison. It is not difficult to understand that the dimensions of the model tend to stabilize. Although the amount of data can be infinitely increased, for a certain area of data there is a certain upper limit. When the amount of data reaches a certain level, the corresponding representative features are basically stabilized. Although it can continue to grow, the growth is extremely slow, at this time, it has reached the approximate gradient optimal. This does not conflict with the infinite extension of the data volume, nor does it contradict the approximation of the great estimation theory.

6. Conclusion

The potential fusion feature model proposed in this paper is a practical problem encountered in the research and application of natural language processing. In this paper, we focus on the problem that the potential link between the features is not fully studied and the characteristics are not deeply integrated with the model, and propose a potential fusion feature model to reduce the complexity of the model and improve the prediction accuracy. Through the theoretical analysis and experimental comparison, the paper demonstrates the validity of our proposed model. Because deep learning has promoted the development of natural language processing, our future research goal is to construct a deep neural network model and further accumulate the amount of source data so that the data model contains a wider range of potential features, to makes the prediction accuracy higher.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Grant 61672138) and the Graduate Science and Technology Innovation Foundation of University of Science and Technology Liaoning(Grant LKDYC201612). We would like to express our gratitude to Sina Microblogging algorithm experts Fu Shengguo and Su Chuanjie for their help in this project.

References

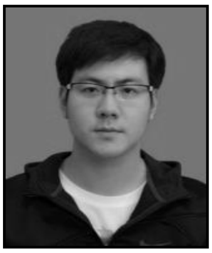
[1] D. Boyd and N. B. Ellison, "Social Network Sites: Definition, History, and Scholarship", Journal of Computer-Mediated Communication, vol. 13, no. 1, **(2007)**, pp. 210-230.
[2] G. Szabo and B. A. Huberman, "Predicting the Popularity of Online Content", Communications of The ACM, vol. 53, no. 8, **(2010)**, pp. 80-88.
[3] K. Lerman and T. Hogg, "Using a Model of Social Dynamics to Predict Popularity of News", International World Wide Web Conferences, **(2010)**.
[4] A. Java, X. Song, T. Finin and B. Tseng, "Why We Twitter: Understanding Microblogging Usage and Communities", Knowledge Discovery and Data Mining, **(2007)**, pp. 56-65.
[5] M. Jenders, G. Kasneci and F. Naumann, "Analyzing and Predicting Viral Tweets", International World Wide Web Conferences, **(2013)**.
[6] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao and Y. Yu, "Collaborative Personalized Tweet Recommendation", International ACM Sigir Conference on Research and Development in Information Retrieval, **(2012)**.
[7] Y. Artzi, P. Pantel and M. Gamon, "Predicting Responses to Microblog Posts", North American Chapter of The Association for Computational Linguistics, **(2012)**, pp. 602-606.
[8] B. Suh, L. Hong, P. Pirolli and H. Chi Ed, "Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network", International Conference on Social Computing, **(2010)**.
[9] A. Kupavskii, A. Umnov, G. Gusev and P. Serdyukov, "Predicting the Audience Size of a Tweet", International Conference on Weblogs and Social Media, **(2013)**.
[10] H. Ma, W. Qian, F. Xia, X. He, J. Xu and A. Zhou, "Towards Modeling Popularity of Microblogs", Frontiers of Computer Science in China, vol. 7, no. 2, **(2013)**, pp. 171-184.
[11] J. Zhang, B. Wang, Y. Xi, X. Liu and Y. Tian, "Micro-Blog Sentiment Summarization Method Based on the Fusion of Multiple Features", Journal of Information Engineering University, vol. 17, no. 2, **(2016)**, pp. 218-224.
[12] J. Cao, J. Wu, W. Shi, B. Liu, X. Zheng and J. Luo, "Sina Microblog Information Diffusion Analysis and Prediction", Chinese Journal of computers, vol. 37, no. 4, **(2014)**, pp. 779-790.
[13] H. Zhang and Q. Liu, "Model of Chinese Words Rough Segmentation Based on N-Shortest-Paths Method", Journal of Chinese Information Processing, vol. 16, no. 5, **(2002)**, pp. 1-7.
[14] C. Wu and S. Wang, "Model of Chinese Words Rough Segmentation based on Bi-gram and N-most-Probability Method", Journal of Computer Application, vol. 27, no. 12, **(2007)**, pp. 2902-2905.
[15] J. Zhou, Z. Zheng and W. Zhang, "Method of Chinese Words Rough Segmentation Based on Improving Maximum Match Algorithm", Computer Engineering and Applications, vol. 50, no. 2, **(2014)**, pp. 124-128.
[16] D. Liu, J. Nie, J. Zhang, X. Liu, C. Wan and G. Liao, "Extracting Sentimental Lexicons from Chinese Microblog: a Classification Method Using N-Gram Features", Journal of Chinese information Processing, vol. 30, no. 4, **(2016)**, pp. 193-205.
[17] G. Salton, A. K. Wong and C. S. Yang, "A Vector Space Model for Automatic Indexing", Communications of The ACM, vol. 18, no. 11, **(1975)**, pp. 613-620.
[18] A. Tsoularis and J. W. Wallace, "Analysis of Logistic Growth Models", Bellman Prize in Mathematical Biosciences, vol. 179, no. 1, **(2002)**, pp. 21-55.
[19] Z.-Q. Liu, G. Rong and Y. C. Feng, "Parallelization of Classification Algorithms Based on SparkR*", Journal of Frontiers of Computer Science and Technology, vol. 9, no. 11, **(2015)**, pp. 1281-1294.

[20] M. Jenders, G. Kasneci and F. Naumann, "Analyzing and Predicting Viral Tweets", International World Wide Web Conferences, **(2013)**.

[21] C. Tan, L. Lee and B. Pang, "The Effect of Wording on Message Propagation: Topic- and Author-Controlled Natural Experiments on Twitter", Meeting of the Association for Computational Linguistics, **(2014)**, pp. 175-185.

Authors

Chengying Chi is a professor of computer science at the University of Science and Technology, Liaoning. Her current research interests are information retrieval, natural language processing, data mining, and distributed database systems.



Shaowei Li is a master student of computer science at the University of Science and Technology, Liaoning. His current research interests are natural language processing, data mining, information retrieval, and machine learning.