Yao *et al. EURASIP Journal on Wireless Communications and Networking* (2017) 2017:166
DOI 10.1186/s13638-017-0950-z

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

A feature selection method based on synonym merging in text classification system

Haipeng Yao[1*], Chong Liu[1], Peiying Zhang[1,3] and Luyao Wang[2]

Abstract

As an important step in natural language processing (NLP), text classification system has been widely used in many fields, like spam filtering, news classification, and web page detection. Vector space model (VSM) is generally used to extract feature vectors for representing texts which is very important for text classification. In this paper, a feature selection algorithm based on synonym merging named SM-CHI is proposed. Besides, the improved CHI formula and synonym merging are used to select feature words so that the accuracy of classification can be improved and the feature dimension can be reduced. In addition, for feature words selected by SM-CHI, this paper presented three weight calculation algorithms to explore the best feature weight update method. Finally, we designed three comparative experiments and proved the classification accuracy is the highest when choosing the improved CHI formula 2, set the threshold α to 0.8 and use the largest weight among the synonyms to update the feature weight, respectively.

Keywords: Text classification, Feature selection, Synonym merging, Feature weights calculation

1 Introduction

With the development of the Internet, the amount of Chinese text information shows an exponential growth trend. How to effectively manage the massive Chinese documents and mine the information contained in the documents has become a critical research problem. Automatic text classification can complete the work of text processing effectively. It also plays an important role in natural language processing (NLP) and data mining.

The most common method used in text classification is the vector space model (VSM). It represents text as a feature vector. The specific process is shown in Fig. 1.

From Fig. 1, we know that the first step in Chinese text classification is to preprocess the text, including word segmentation, part of speech tagging, and removal of stop words. The purpose is to remove the useless words and only leave the nouns, adjectives, and verbs that contain category information. After this, the text can be represented as a vector to form VSM. Then, we use the feature selection method to select the feature words that can symbolize the text categories, and merge the synonym to reduce dimensions. Next, TF-IDF [1] method is used to calculate the weight of each feature of each text to transform the text into a feature vector. Last but not least, by using the Bayesian classifier to train the sample data, we can get the final text classifier.

Feature selection is the most important step because the selected feature words directly affect the accuracy of the classifier. In VSM, the best feature selection method is χ^2 statistics (CHI) [2, 3]. But the defect is high-dimensional feature vectors selected by CHI may cause dimension disaster. The writer consider to merge the synonyms among the feature words selected by CHI so that the dimension of feature space can be reduced. Then, in the next step, an improved TF-IDF method is used to calculate the feature weights for each word to generate the feature vector of each text. This paper mainly studies the influence of feature selection and synonym merging on the accuracy of classification in automatic text classification. Synonym merging can reduce the dimension of the feature space and improve the classification performance. The study of feature selection algorithm and synonym merging has a strong practical significance. The main contributions of this paper are as follows:

*Correspondence: yaohaipeng@bupt.edu.cn
[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Haidian District Xitucheng Road 10, Beijing 100876, People's Republic of China
Full list of author information is available at the end of the article

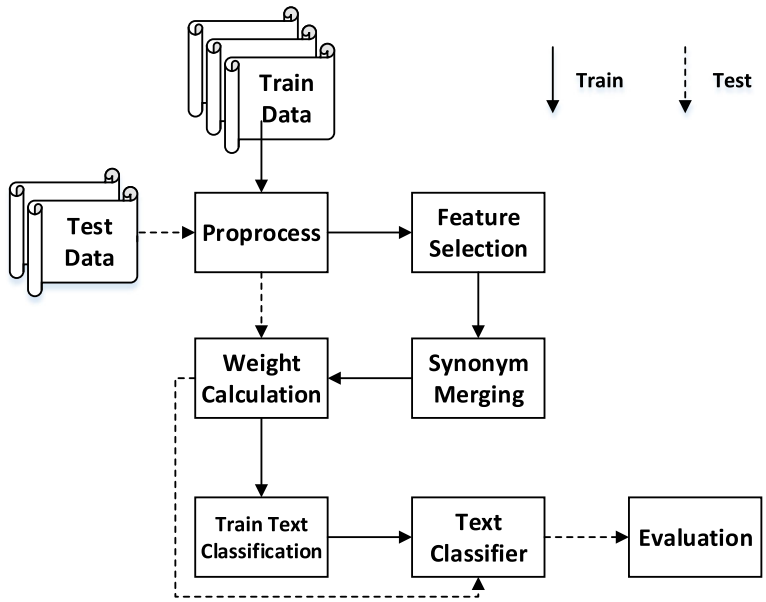

Fig. 1 Text classification model framework

1. We presented a new feature selection algorithm named SM-CHI based on an improved CHI [4] formula and synonym merging to achieve efficient feature selection and dimension reduction.
2. We found that the original CHI formula multiplied by a log term has the best classification performance comparing with the original CHI and two improved CHI algorithms [4, 5].
3. The choice of thresholds α ($0 \leqslant \alpha \leqslant 1$) is critical. Only the most similar feature words will be merged when α is close to 1, so we use grid search method to find the optimal α. The result show that the classification accuracy is highest when α is equal to 0.8.
4. We proposed three improved weight calculation methods based on TF-IDF. The experimental results show that using the maximum value of the synonym group as the feature weight is the best way.

The organization of this paper is as follows. Next section discusses some related works. The text classification system based on semantic similarity is introduced in Section 3. We present the details of SM-CHI and the corresponding weight calculation methods in Section 4. In Section 5, we show and discuss the experimental results. We conclude our work and describe future work in Section 6.

2 Related works

Text classification has been widely used in the classification and labeling of news and web pages. There are many works on text classification. The classification methods based on synonym merging have appealled much attention in the past 2 years. Next, we briefly introduce some of the existing research results.

2.1 Classification model

Nowadays, most of the text classification methods are based on VSM where the texts are represented in the form of (feature vector, label). In this way, we can transform each text into feature vectors to construct the training and testing dataset. It is a long existing problem of machine learning to train a classifier for text classification. Many machine learning algorithms can be used as the classifier, such as Naive Bayes Method [6], k-Nearest Neighbor [7], Decision Trees [8], Support Vector Machines (SVM) [9], Markov Model [10], and Neural Networks [11]. KNN as one of the best classifiers in VSM is simple, effective, non-parameter. So we choose it as the classification method in this paper.

2.2 Feature selection algorithm

In order to construct a valid classifier, feature words selection is another main problem. Common feature selection methods include document frequency (DF), mutual information (MI) [12], a χ^2 statistics (CHI), information gain (IG) [6, 13], and term strength (TS) [14, 15]. The works in [2, 3] show that CHI is the best feature selection method through contrast experiments. However, CHI feature selection method also has shortcomings [16]. For example, the CHI value of the high-frequency words is very high, but they have no significant contribution to

class distinctions. Therefore, the authors of [4, 5] presented two improved CHI formulas from different perspectives in order to make up for the lack of the original CHI method.

2.3 Synonym merging

With the development of text classification, some researchers start to propose a text classification system based on synonym merging to improve the accuracy of classification. The word similarity calculation methods based on "Tong YiCi Cilin" and "HowNet" were proposed in [17, 18] respectively. The work in [19] proposed a text feature selection method based on "TongYiCi Cilin" to reduce data's feature dimensions while ensuring data integrity and classification accuracy. A semantic kernel is used with SVM for text classification to improve the accuracy in [20, 21]. What is more, there are some other work in [22–24] that presents some excellent ideas, which is worth learning and reference when we are dealing with large-scale text classification. They can help us to speed up the calculation through big data technology.

The model proposed in this paper is a text classification model based on synonym merging, named SM-CHI. The difference with [19] is that we merge synonyms after feature selection based on CHI and we propose three improved weighting method for the merged feature words.

3 Text classification model based on semantic similarity

In this section, we mainly introduce the text classification model based on semantic similarity. Wherein, Section 3.1 describes the feature selection method based on χ^2 statistic. Section 3.2 introduces the method of synonym merging; Section 3.3 presents the traditional weight calculation method, TF-IDF.

3.1 Improved feature selection algorithm

In text classification, a feature word and its category tend to obey the CHI formula. Higher CHI value implies that a feature word has stronger ability to identify a category. The CHI value of word is calculated as follows [3]:

$$\chi^2(t,c) = \frac{N * (AD - BC)^2}{(A+C)(A+B)(B+D)(C+D)} \tag{1}$$

where N is the size of the training set; A is the number of documents that belong to class c and contain the word t; B is the number of documents that do not belong to class c but contain the word t; C is the number of documents that belong to the class c but do not contain the word t; and D is the number of documents that do not belong to class c and do not contain the word t.

Although the CHI formula has a relatively good performance in text classification, it also has some shortcomings [16]. First of all, high-frequency words that appear in all categories have higher CHI values, but they do not make much sense for class distinctions. Secondly, the CHI formula only considers the appearance of a word but not the frequency of the word in a document. Therefore, CHI formula also has "low frequency words flawed". For example, assuming word $t1$ appears in 99 documents, each appears 10 times; word $t2$ appears in 100 documents, each appears one time; obviously $t2$ has a higher CHI value, but in fact $t1$ is more representative for this category. There are many studies amended for its defects. The work in [4] proposed the multiplication by a log entry based on the original CHI to reduce the CHI value of high-frequency words. The formula is as follows:

$$chi\_imp\_1 = \log\left(\frac{N}{A+B}\right) * \chi^2(t,c) \tag{2}$$

where $A + B$ represents the number of documents that contain word t and N represents the total number of documents. In this case, the CHI value of the high-frequency words that appear in all categories are close to zero so that they would not be selected as a feature word.

In addition, the work in [5] has made a corresponding improvement to the word frequency, which is multiplied by term $\beta(t,c)$ on the basis of the original CHI formula. The calculation formula is as follows:

$$chi\_imp\_2 = \beta(t,c)\chi^2(t,c) \tag{3}$$

where $\beta(t,c)$ is calculated as follows:

$$\beta(t,c) = \frac{tf(t,c)}{\sum_{i=1}^{m} f(t,c_i)} \tag{4}$$

In the formula, m is the total number of categories and $tf(t,c)$ is the frequency of the word t in the category c.

3.2 Synonym merging algorithm based on "Tong YiCi Cilin"

Some of the feature words selected by CHI formula may be the same or have similar meaning. They have the same effect on class distinctions. If the synonym are merged, not only the classification accuracy will be improved, but also the dimension of the feature space can be reduced so that the efficiency of the algorithm can be improved. For example, "GanMao", "ZhaoLiang", "ShangFeng" are the synonym of "Cold" in Chinese. If the "Health Care" category articles contain these words respectively, then the feature words for the text classification contain too much redundant information. We use the method of synonym merging to deal with it.

In this paper, we use the "Tong YiCi Cilin" provided by Harbin Institute of Technology as the method of word similarity calculation [17]. Its structure has five layers. You can easily calculate the similarity between the two terms. The structure of "Tong YiCi Cilin" is shown in Fig. 2.

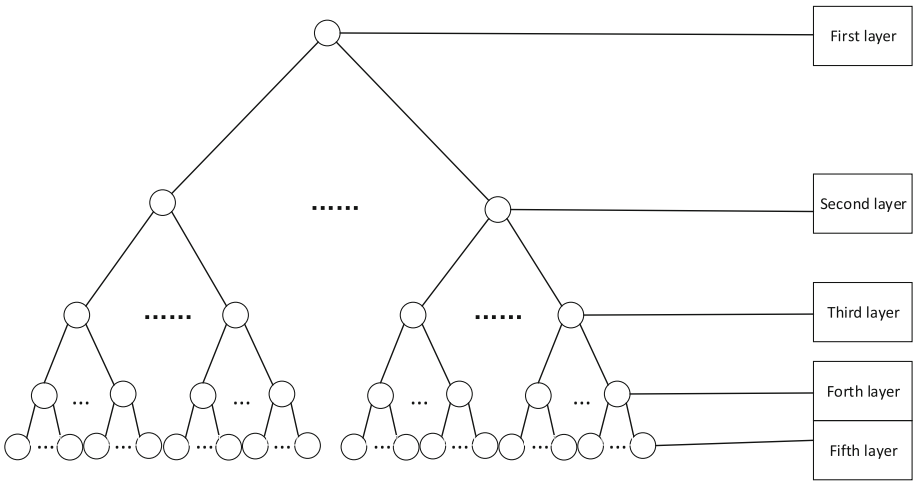

Fig. 2 Hierarchy structure of "Tong YiCi Cilin"

The concrete similarity calculation method is introduced in detail in [17]. When the similarity of two words is greater than threshold α, they will be regarded as a pair of synonym to merge. The optimal value of α will be discussed later in the experiment. In addition, all merged synonyms are stored in a list. Nested lists are used to store feature words so that all synonym information remains in the feature vector. To calculate the feature vector of each document, we propose three improved methods which will be discussed in Section 3.3.

3.3 Weight calculation method

Traditional TF-IDF weight calculation formula [1] is as follows:

$$weight_{t,d} = tf_{t,d} + idf_t \tag{5}$$

This formula represents the weight calculation method for word t in document d. Here, $tf_{t,d}$ denotes the frequency of occurrence of word t in d, and idf_t denotes the anti-document frequency of t, which is used to quantify the distribution of t in the training set. If n is used to denote the number of documents which contain t in the training set, the calculation formula of idf_t is as follows:

$$idf_t = \log\left(\frac{N}{n}\right) \tag{6}$$

As we mentioned above, the TF-IDF method is used to calculate the weight of each feature word in each text.

4 Algorithm description

4.1 Feature selection method based on the synonym merging

In this section, we introduce the feature selection algorithm (SM-CHI). This method firstly selects candidate feature words based on an improved CHI formula, and then merges synonyms to re-select those feature words that can represent the categories better and reduce dimension. The method is represented as the following formula:

$$SM - CHI = LF(t) * CHI(t) * SM(t) \tag{7}$$

where $LF(t)$ denotes whether the word t exists in the word bag or not, and is mainly decided according to the part of speech and stopping words. If the word t is a stopping word and in the part of speech that does not belong to verb, noun, and adjective, $LF(t) = 0$, otherwise $LF(t) = 1$. $CHI(t)$ represents the CHI value of the word t and is calculated by Eq. (2). $SM(t)$ indicates whether the word t contains synonym. If yes, it needs to merge all of its synonyms.

Firstly, all the texts in the training set are preprocessed, including Chinese word segmentation, part-of-speech tagging, and discarding stop words. The remaining words constitute the word bag of the training set. Secondly, we calculate the CHI value of each word. Choose the first 200 words from each category to form candidate sets of feature words. Note that the characteristic words selected for each category may be duplicated. The candidate set is stored using the HashSet (a data structure) and the de-emphasis is performed. After obtaining the candidate set, the similarity between each word is calculated according to "Tong YiCi Cilin" and threshold is set to α. The synonym merging is performed only when the word similarity is greater than α. We will experimentally determine the optimal value of hyper-parameter α. The pseudocode of SM-CHI is shown in Algorithm 1.

4.2 Improved method for calculating eigenvalue weight

In the scene of SM-CHI feature selection method presented in this paper, the traditional TF-IDF formula has some drawbacks. For the features after synonym merging,

Algorithm 1 SM-CHI feature selection algorithm

```
Input:a training set D
Output:a feature space F
(Initialization) A, B, TFIDF can all be a null dict
for each file in D:
    word_list=file.process()
    for word in word_list:
        A[file.class][word] += 1
        TFIDF[file.class][file.num][word] +=1
end for
Calculate B in the CHI formula from A
for cla in
    for word in cla:
        Calculate the CHI value according to formula 2
    Selects the first 200 words as the feature
end for
Combine features of the 9 categories to obtain word_features
for word1, word2 in word_features:
    sim=calcWordsSimilarity(word1, word2)
    if sim > α:
        Merge word1 and word2
end for
```

the original weight calculation formula will cause "unfairness". Because the merged feature words are stored in the nested list, so which word among them will be regarded as the feature is a question. For this problem, we present the following three solutions:

- Sum the weights of all items up in the feature list of each dimension as the weight of the list;
- Take the largest weight among the synonym as the weight value of the feature;
- Multiply the first item by 1.1 for times of the number of items in the feature list.

5 Experiments and results

In this section, three groups of experiments are designed to evaluate the performance of three CHI formulas, the optimal threshold α for synonym merging, and the performance of feature weight update method. We use the whole news data set from Sogou Lab [25] to test the accuracy of the experiment.

5.1 Performance evaluation and data set

The standard precision rate P, recall rate R, and $F1$ score are used to measure the classification performance. For the i_th category, the formula is as follows [26]:

$$P_i = \frac{TP}{TP+FN}, R_i = \frac{TP}{TP+FP}, F1_i = \frac{2*P_i*R_i}{P_i+R_i} \quad (8)$$

where TP is the number of documents correctly classified as class i, FP is the number of documents classified as class i but not actually i, and FN is the number of documents that is not classified as class i but is actually class i.

This article will use the whole network news data set provided by Sogou Lab to test our experiments. The corpus includes nine kinds of news types, such as Automobile, Finance, and IT. Each category contains thousands of documents. In this experiment, each category takes 400 documents, of which 280 are training set and 120 are test sets. Therefore, the training set contains 2520 documents, and the test set includes a total of 1080 documents.

The preprocessing module uses a third-party library for python, named jieba, to complete the work of word segmentation, part of speech tagging, and discarding of stop words. In addition, we use Naive Bayesian classifier provided in Python's NLTK library as the classifier.

5.2 Experiments and results

In this section, the following three groups of experiments are carried out to test the three innovation points of SM-CHI with control variable method . In Experiment I, we test the three feature selection algorithms without using synonym merges. In Experiment II, we use the first improved CHI method to select features and use grid search method to find the optimal threshold α. On the basis of Experiment I and II, we designed Experiment III to find the best weight update method.

5.2.1 Experiment I

In order to test the effect of three kinds of feature selection methods described in section 3.2, we conducted the following experiments. But we did not use synonym merging here. The results of experiment are shown in Fig. 3:

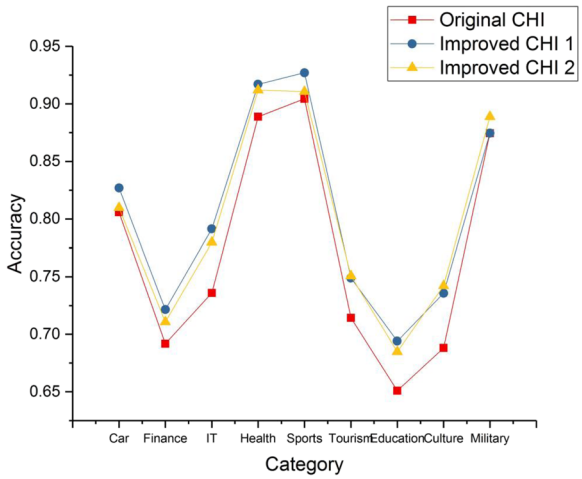

Fig. 3 Classification results comparing three kinds CHI formula

From the results, we can see that the two improved CHI formulas have a great effect on enhancing the value of F1 score of each category as compared to the original CHI formula, which means that the improved CHI formulas can select more representative words. They both make some improvement based on the original CHI. In addition, when the two improved CHI formulas are compared, the first improved method has a slight advantage, showing a better discrimination effect in the preceding categories. The result also shows that the log term successfully suppresses the CHI values of the high-frequency words appearing in all classes, which achieves relatively good results. Therefore, we will use the first improved CHI formula as our base feature selection method in the fellow experiment.

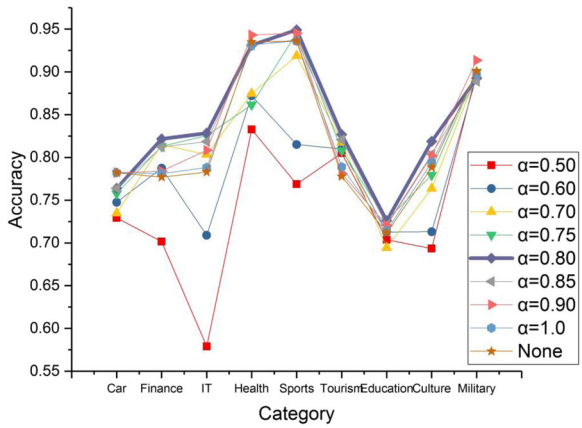

Fig. 4 F1 score of each category when α takes different values

5.2.2 Experiment II

In order to select the appropriate threshold α for synonymy merging, we designed the following experiment. The first improved CHI formula was used for feature selection, and the range of α is set to [0.5, 0.6, 0.7, 0.75, 0.8, 0.85, 0.9,1.0]. A total of nine experiments are conducted, including a comparative experiment that did not use synonym merging. The experimental results are shown in Table 1 and Fig. 4.

From the results in Figs. 4 and 5, we can draw the conclusion that the classification accuracy is the highest when $\alpha = 0.8$ and worst when $\alpha = 0.5$. The use of synonym merging improved classification accuracy by approximately 3 percentage points compared to use CHI only. By specific analysis of each category, we found that when we use synonym merging, only the first category has a low accuracy compared to no synonym merging. The reason is that the eigenvectors after synonym merging have reduced the text discrimination degree of the "car" category.

5.2.3 Experiment III

Based on the previous two experiments, we designed the following experiment to select the optimal weight update method introduced in Section 3.3. We designed three experiments, all with the first improved CHI formula and a threshold of 0.8. The experimental results are shown in Fig. 6:

According to Fig. 6, it can be seen that the classification accuracy of method 1 is the lowest. The reason is that this method adds the weights of all the synonymy words as the weight of the feature, but a word and its synonym words appear in more than one category. This simple superposition will make the feature differentiate the category worse. In contrast, methods 2 and 3 use the combined synonym as a one-dimensional feature and achieve better classification results and F1 scores. In contrast, method 2 is more effective which shows that the maximum value of the synonym is a better method because it can represent the maximum abilities of all synonyms to differentiate the text categories. By multiplying the power of 1.1 by the n, method 3 incorrectly increases the ability of the feature to distinguish text categories.

6 Conclusions

In this paper, we propose the SM-CHI feature selection method based on the common method used in Chinese text classification. This method mainly considers

Table 1 F1 score of each category when α takes different values

	Car	Finance	IT	Health	Sports	Tourism	Edu	Culture	Mil
$\alpha = 0.50$	0.729	0.701	0.579	0.833	0.769	0.805	0.704	0.693	0.900
$\alpha = 0.60$	0.747	0.787	0.709	0.872	0.815	0.810	0.712	0.713	0.895
$\alpha = 0.70$	0.734	0.814	0.803	0.874	0.919	0.817	0.694	0.764	0.896
$\alpha = 0.75$	0.757	0.813	0.825	0.861	0.946	0.808	0.726	0.779	0.900
$\alpha = 0.80$	0.763	0.821	0.828	0.931	0.949	0.827	0.726	0.819	0.903
$\alpha = 0.85$	0.763	0.812	0.818	0.931	0.937	0.821	0.721	0.804	0.889
$\alpha = 0.90$	0.782	0.783	0.808	0.943	0.945	0.781	0.720	0.802	0.913
$\alpha = 1.0$	0.782	0.781	0.788	0.932	0.936	0.789	0.714	0.794	0.897
None	0.782	0.776	0.783	0.935	0.936	0.778	0.711	0.788	0.901

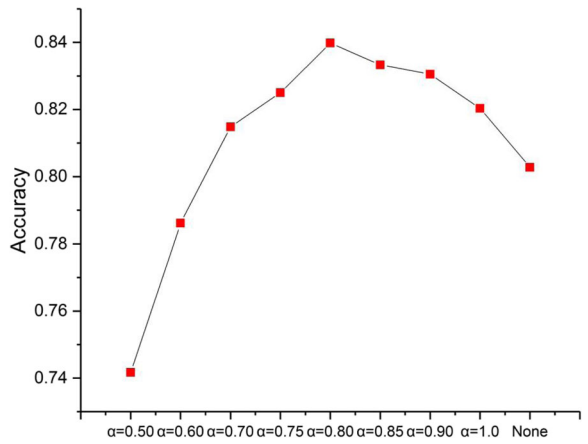

Fig. 5 Total accuracy when α takes different values

part of speech tagging, improved CHI formula and synonym merging. In addition, this paper proposes an update method for calculating the weight of feature words after synonym merging to obtain a more accurate vector representing of the text for classifier processing. The results of the experiment proves that the feature dimension can be reduced and the accuracy and effectiveness of text classification can be improved at the same time with this method.

In the future, we will focus on using synonym similarity calculation method based on "HowNet" instead of "Tong YiCi CiLin", because "HowNet" uses more than 1500 generics to build a unique knowledge description form and rich lexical semantic knowledge, so that we can more accurately calculate the similarity of words. What is more, SVM and neural networks have become the mainstream classifier in the text classification system because of its high accuracy, so we will use SVM or neural network instead of Naive Bayesian classification.

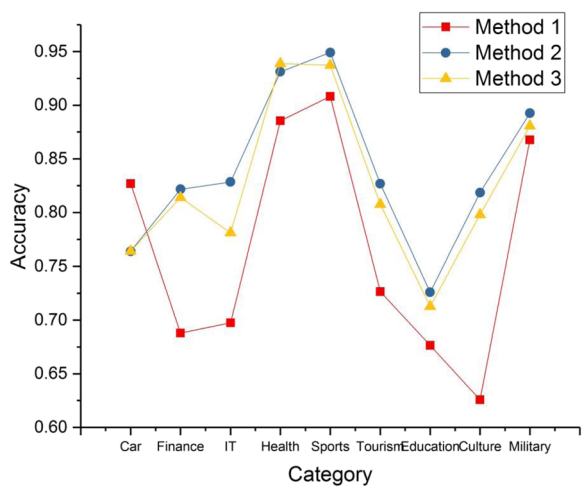

Fig. 6 Three kinds of special effects classification weight value updating method of comparison

Acknowledgements
This work is supported in part by the Shandong Provincial Natural Science Foundation, China (Grant No. ZR2014FQ018), in part by the BUPT-SICE Excellent Graduate Students Innovation Fund, in part by the National Natural Science Foundation of China (Grant No. 61471056), and in part by the China research project on key technology strategy of infrastructure security for information network development. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Authors' contributions
The idea was arisen from the discussions between HY and PZ. CL did the simulation and code implementation and wrote the Chinese version of the paper with the guide of PZ. HY wrote the Abstract and Conclusions. LW help us to translate the paper into English and made a lot of changes. All authors read and approved the final manuscript.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Author details
[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Haidian District Xitucheng Road 10, Beijing 100876, People's Republic of China. [2]Advanced Innovation Center for Future Internet Technology, Beijing University of Technology, Chaoyang District Pingleyuan 100, Beijing 100124, People's Republic of China. [3]College of Computer & Communication Engineering, China University of Petroleum (East China), Changjiang West Road 66, Qingdao 266580, China.

Received: 31 July 2017 Accepted: 27 September 2017
Published online: 05 October 2017

References

1. wikipedia, tf-idf . https://en.wikipedia.org/wiki/Tf%E2%80%93idf. Accessed 01 Sept 2017
2. HT Ng, WB Goh, KL Low, in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. Feature selection, perceptron learning, and a usability case study for text categorization (ACM, Philadelphia, 1997), pp. 67–73
3. Y Yang, JO Pedersen, in *Fourteenth International Conference on Machine Learning*. A comparative study on feature selection in text categorization (Morgan Kaufmann Publishers Inc., 1997), pp. 412–420
4. GC Feng, S Cai, in *Fourth International Conference on Computer, Mechatronics, Control and Electronic Engineering*. An Improved Feature Extraction Algorithm Based on CHI and MI (ICCMCEE, 2015)
5. Y Tang, T Xiao, in *International Conference on Computational Intelligence and Software Engineering*. An improved χ^2 (chi) statistics method for text feature selection (IEEE, 2009), pp. 1–4
6. DD Lewis, M Ringuette, in *Third Annual Symposium on Document Analysis & Information Retrieval*. A comparison of two learning algorithms for text categorization (ISRI, 1994), pp. 81–93
7. Y Yang, An evaluation of statistical approaches to text categorization. Inf. Retr. J. **1**(1), 69–90 (1999)
8. E Velasco, LC Thuler, CA Martins, LM Dias, VM Gonçalves, Automated learning of decision rules for text categorization. Acm Trans. Inf. Syst. **12**(3), 233–251 (1994)
9. T Joachims, *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*. (Springer Berlin, Heidelberg, 1998)
10. J Bhimani, N Mi, M Leeser, Z Yang, in *IEEE International Conference on Cloud Computing*. Fim: Performance prediction model for parallel computation in iterative data processing applications (IEEE, 2017)
11. E Wiener, J Pedersen, AS Weigend, A neural network approach to topic spotting. Proc. Fourth Ann. Symp. Document Anal. Inf. Retr. (SDAIR). **92**(3), 482–487 (1995)
12. KW Church, P Hanks, Word association norms, mutual information, and lexicography. Comput. Linguist. **16**(1), 22–29 (1989)
13. JR Quinlan, Introduction of decision trees. Mach. Learn. **1**(1), 81-106

14. WJ Wilbur, K Sirotkin, The automatic identification of stop words. J. Inf. Sci. **18**(1), 45–55 (1992)
15. Y Yang, in *International ACM SIGIR Conference on Research and Development in Information Retrieval*. Noise reduction in a statistical approach to text categorization (ACM, 1995), pp. 256–263
16. T Dunning, Accurate methods for the statis15 of surprise and coincidence. Linguist. 74 Comput. Dirk Geeraerts Stefan Grondelaers. **19**(1), 61–74 (1993)
17. J Tian, W Zhao, Words similarity algorithm based on tongyici cilin in semantic web adaptive learning system. J. Jilin University. **28**(06), 602–608 (2010)
18. SJ Li, *Word Similarity Computing Based on How-net, The third Chinese mandarin semantics seminar*, (Taibei, 2002)
19. YH Zheng, DZ Zhang, A text feature selection method based on tongyici cilin. J. Xiamen University. **51**(2), 200–203 (2012)
20. B Altınel, MC Ganiz, B Diri, A corpus-based semantic kernel for text classification by using meaning values of terms. Eng. Appl. Artif. Intell. **43**(C), 54–66 (2015)
21. B Altınel, B Diri, MC Ganiz, A novel semantic smoothing kernel for text classification with class-based weighting. Knowl. Based Syst. **89**, 265–277 (2015)
22. J Wang, T Wang, Z Yang, Y Mao, N Mi, B Sheng, in *International Conference on Computing, NETWORKING and Communications*. Seina: A Stealthy and Effective Internal Sttack in Hadoop Systems (IEEE, 2017)
23. Z Yang, J Wang, D Evans, N Mi, in *International Workshop on Communication, Computing, and NETWORKING in Cyber Physical Systems*. Autoreplica: Automatic Data Replica Manager in Distributed Caching and Data Processing Systems (IEEE, 2016)
24. J Wang, T Wang, Z Yang, N Mi, B Sheng, in *IEEE International PERFORMANCE Computing and Communications Conference*. eSplash: Efficient Speculation in Large Scale Heterogeneous Computing Systems (IEEE, 2016)
25. sogou, Sogou data. http://www.sogou.com/labs/resource/ca.php. Accessed 01 Sept 2017
26. S Qin, J Song, P Zhang, Y Tan, in *International Conference on Fuzzy Systems and Knowledge Discovery*. Feature selection for text classification based on part of speech filter and synonym merge (IEEE, 2015), pp. 681–685