# Research and Improvement of feature words weight based on TFIDF Algorithm

Aizhang Guo
Qilu University of Technology
Jinan250353, China
gaz@qlu.edu.cn

Tao Yang
Qilu University of Technology
Jinan250353, China
yangtao8812@163.com

***Abstract*—With the development of cloud era, more and more people have been attracted by Big data. More and more applications involve large data. Analysis methods of large data is particularly important. This paper mainly analyzes and research feature words weight which are used in unstructured data classification of big data. Firstly, we combine the traditional feature words weight calculation method and analyze the shortcoming of traditional TF-IDF algorithm, It doesn't think about feature words distribution. It can lead that some feature words weight which don't have strong discrimination have heavier weight. Aiming at the shortage of TFIDF algorithm, combining with practical effect to text classification, this paper modify traditional TFIDF algorithm formula, excluding the inner impact to disturb characteristic, adding the concept of intra-class dispersion, presenting a new TFIDF algorithm. In the experiment, experimental data comes from People news about the financial, military, entertainment and sports four categories, respectively calculating test value by using the traditional TFIDF algorithm and improved TFIDF algorithm. Results show that improved TFIDF algorithm has higher accuracy than traditional TFIDF algorithms.**

***Keywords- TFIDF algorithm;Text classification; Feature selection; Feature weighting***

## I. INTRODUCTION

With the rapid development of network technology, vast amounts of information resources exist in the form of text. People urgently hope to find interesting contents from large information quickly and efficiently. Text classification is as an important research direction of information processing, it is a common method for solving text information discovery problems. However, a large number of documents doesn't mark keywords. If people mark the keywords of these documents , it is difficulty and will spend much time, we urgently need to carry on extracting keywords automatically and introduce concept. People through a certain algorithm to divide text words which exist in network, thus forming the feature words, according to certain algorithms, give each feature word different weights. There are many ways to calculate weights, including Boolean functions, frequency functions, logarithmic function, entropy function and TFIDF function and so on. TFIDF algorithm is relatively simple and has a higher accuracy rate and the recall rate, so it has been the concern of many related researchers. The main idea of TDIDF algorithm is: the more times a word appears in a sentence, then the greater contribution where words describe the meaning of the sentence; the more documents where words appear, then the word for a contribution of the document should be smaller. By TFIDF algorithm, people can quickly locate the content of documents .

Therefore, how to divide effective feature words can have high weights and reduce number of zero weight feature words is core of TFIDF algorithm. To enhance the speed of browsing the article, the paper analyzes shortcoming of traditional TFIDF algorithm massive data classification, help people find the text quickly and accurately the information they need, this paper analyzes the advantages and disadvantages of traditional TDIDF algorithms, researching and studying of existing research results at domestic and foreign, combing with practical implications to text classification, improving the traditional TFIDF algorithm. Finally, we present a new TFIDF algorithm.

## II. DOMESTIC AND FOREIGN RESEARCH STATUS AND ACHIEVEMENT

Jones K S proposed IDF idea[1] firstly in 1972,He pointed out that: in a set of documents, if the higher feature items appear in all the document , less information entropy it contains, the corresponding weight should be lower; if a certain feature concentrated on a small amount of documentation , then it will contain higher information entropy and corresponding weights should be higher. In 1973, Salton combined the idea of K S JONES and presented a TFIDF (Frequency &Inverse Documentation Frequency Term) algorithm in paper [2] . Since then, he has repeatedly demonstrated the effectiveness of the algorithm in information retrieval [3], and in 1988 he put the feature words and weight into literature retrieval, and discusses the experimental results [4], and then he comes to the conclusion that the TFIDF algorithm has the following ideas: if the frequency of a word or phrase in the chapter of TF is high, and in the other article rarely appear, think the word or phrase has good ability to distinguish and suitable for classification; the wider scope where a word appears in a document , the lower attributes that it distinguishes the document content is low (IDF). In 1999 Roberto Basils proposed an improved TF IWF IWF algorithm [5], the algorithm improves the weight of feature words when

**978-1-4673-9194-8/16/$31.00 ©2016 IEEE**

documents have a lower appearing frequency ,It is good to distinguish mufti documents. According to the number of different types of documents, Bong Chih How and Narayanan K may are orders of magnitude gap and puts forward the category Term Descriptor (CTD) to improve TFIDF[6] In 2004 . It solves the bad influence to the TFIDF algorithm from different categories documents number .

In China, many research scholars research and improve TFIDF algorithm and made a lot of significant results. In 2006, [7] In order to solve the distribution characteristic items between classes and inner classes, ZHANG Yu-fang etc modified the TFIDF formula. The algorithm takes into account the inner distribution of characteristics items. It improved feature items weights which mostly appeared in a certain category and less in other types of text content. It can accurately distinguish those documents. Paper [8] analyzed the TFIDF and TFIWFIWF formulas and used n times variance root word weights to adjust the weights to the frequency of reliance from the angle from TF, it introduced the variance items from the angle of IWF. Paper [9] elaborated development of TFIDF, improved 1TFIDF algorithm, analyzed and summarized its adaptability, got the analysis results through related improvements through and provided a reference for people who learn TFIDF later. Paper [10] proposed an algorithm which generates a central vector for each text, when it happens a new text, calculating the distance of the text to center of the text to determine the text belongs to which type of text. Facing about the deficiency of Frequency Anti document frequency (TFIDF) algorithm defects, paper [11] proposed a combination of inter-class and intra-class dispersion entropy class discrimination algorithm. The algorithm takes into account the characteristic frequency of the inter-class features words distribution and other practical factors to improve the accuracy of distinct texts. Paper [12] analyzed that the original TFIDF algorithm. It did not take into account the distribution of characteristic words in the category, it results that the algorithm's accuracy is not high, the proposed new weight increase a class scatter to observe words in the distribution of selected features within the class, through experimental analysis to verify the effectiveness of the algorithm.

According to the above analysis for the research , I feel that there exists some flaws in formulas modifications and calculating inner-class degree. Therefore, this paper analyzed the results of previous studies, improved its shortcomings and presents new TFIDF algorithms.

## III. IMPROVED TFIDF ALGORITHM AND IMPROVED

Document mainly consists of connection words, so in order to classifying the document, we must select representative words in all documents. In documents, the importance of the documents is called word weight. Bigger the weight is, more able to represent the theme of the document. In TFIDF algorithm, TF represents the frequency which appears in the document d, IDF documents mainly reflects the distribution of documents which contain character words in the total documents. The main idea of TFIDF is that the higher frequency a word appears in an article, less rarely appear in other articles. We are considered that it has a good ability to distinguish words.

### A. The traditional TFIDF algorithm

In TFIDF, TF is the number of feature items which appear in the document, IDF is anti-document frequency. The formula is:

$$w_{ij} = tf_{ij} \times idf_i = tf_{ij} \times \log\left(\frac{N}{n_i} + 0.01\right)$$

Taking into account that the length of document contents will affect the weight calculation, we use normalized processing and get the following formula:

$$w_{ij} = \frac{tf_{ij} \times \log\left(\frac{N}{n_i} + 0.01\right)}{\sqrt{\sum_{i=1}^{N} tf_{ij}^2 \times log^2\left(\frac{N}{n_i} + 0.01\right)}}$$

Wherein, $tf_{ij}$ is the number of features appearing in the document, $idf_i$ refers to the reciprocal of the feature item, N is the total number of documents, $n_i$ is a number of documents that feature items appear.

### B. The shortcomings of the traditional TFIDF algorithm

*1) IDF does not take into account the distribution information between the feature words*

If m is number of documents which contains word t in a certain class, k is total number of documents in other classes, It is clear that number of documents contain t is n = m + k, m increases with increase of n, according to IDF formula can get little IDF value. It indicates that the discrimination ability of term t is not strong. But in fact, m is larger, which meant that word t frequently appears in word documents, it shows a good representative of t text feature class, it should be given a higher weight and selected as text feature words. On the other hand, although the number of documents n containing t is small, but if it is evenly distributed among the various categories, these feature words are not suitable for classification, it should be given less weight, in accordance with the traditional TFIDF algorithm to calculate IDF value, the IDF value is great.

*2) TFIDF feature does not consider the case of the word incomplete classification*

If m is number of documents which contains word t in a certain class, k is total number of documents in other classes, It is clear that number of documents contain t is n = m + k, m increases with increase of n, according to IDF formula can get little IDF value. It indicates that the discrimination ability of term t is not

strong. But in fact, m is larger, which meant that word t frequently appears in word documents, it shows a good representative of t text feature class, it should be given a higher weight and selected as text feature words. This is one aspect which is not considered IDF feature words distribute in between class distribution; on the other hand, although the number of documents n containing t is small, but if it is evenly distributed among the various categories, these feature words are not suitable for classification, it should be given less weight, in accordance with the traditional TFIDF algorithm to calculate IDF value, the IDF value is great.

*3) Classify the text vectors TFIDF feature does not consider the distribution of information within the word class*

Concentrated in a category different feature item similarly, evenly distributed within the feature item weight should be higher than the uneven distribution. See Ref [12].

### C. Improved algorithms TFDIF

*1)* According to IDF does not take into account the distribution of information between feature items, we have modified IDF formula to increase the weights of those feature items frequently appear in a class, reducing feature item weight which evenly distributed between the different classes. At the same time, we have introduced a training set and added the parameter K (depending on the type of document to adjust size of parameter K ). Improved IDF algorithm is as follows:

$$\mathrm{IDF} = \mathrm{logn} \times \log\left(\frac{N}{n+k} + 0.01\right)$$

（Where $n \in N^+$， and $n + k \neq 0$）

Among them, the total number of text documents is N, the number of documents that contain feature words is n, k is an arbitrary parameter.

When the number of documents containing feature words n is very small and tends to 1, it shows that distinguish ability is poor, it should have very little weight, in the IDF formula, when n tends to 1, IDF tends to 0, just to meet; when the number of documents that contain features words n is very large, and tend to be N, shows that the distinguishing feature words ability is poor, should be given very little weight, in the IDF formula, when n tends to n, IDF tends to 0, just to meet; when the number of documents that contain feature words n increases, distinguish ability for the characteristic words to discriminate should be gradually increased, when n reaches a certain value , the words documents distinguish ability decrease with increase of n ,in IDF formula, IDF first increased and then decreased, and when n tends to 1 and n tends to n, IDF tends to 0, just to meet the requirements.

When classifying to different types of documents, the same characteristic words should have different weights, so we added a variable constant k to adjust selected feature words weights , through the training set to get most suitable k value and give feature words more accurate weights to improve text classification accuracy.

*2)* Because the IDF did not take into account the distribution information of characteristics within the class, I refer to calculation formula of feature words of[12] and modify it, increase class scatter CD, refer to statistical standard deviation formula and get the following the calculation formula about class scatter within the CD:

$$\mathrm{CD} = \frac{\sqrt{\left[\sum_{j=1}^{n}\left(tf_{ij} - \overline{tf}\right)^2\right]}}{\frac{k-1}{\overline{tf}}}$$

Wherein, $\overline{tf_{ij}} = \frac{1}{n}\sum_{j=1}^{n} tf_{ij}$ ; k is the total number of documents within the category, $tf_{ij}$ represent the number of feature words appear in the chapter of j; $\overline{tf}$ is the average number of feature words appearing in the class each document. If the feature appears in only a few words of the document, indicating that the classification ability is poor, dispersion CD within the class can get the value closing to 1; if TF value of each document in the within class documentation are roughly equal, indicating that classification ability is good, within class scatter can tends to 0 .

Since the distribution of items between the inner have great effects on text classification, so when calculating weight of feature words, IDF should achieve greater value; Distribution of items between the inner have little effects on text classification, adjusting weight of text feature words lightly, so when the right word calculating feature, we should get smaller CD value. Because taking into account positive proportions of feature words classification capability and degrees of within classes, we use formula (1-CD) to stand by constructing feature words weight. So we finally get TFIDF algorithm formula is as follows:

$$tw_{ij} = tf_{ij} \times \mathrm{logn} \times \log\left(\frac{N}{n+k} + 1\right) \times \left(1 - \frac{\sqrt{\left[\sum_{j=1}^{n}\left(tf_{ij} - \overline{tf}\right)^2\right]}}{\frac{k-1}{\overline{tf}}}\right)$$

The above formula is normalized and get the following formula:

$$w_{ij} = \frac{tf_{ij} \times \log n \times \log\left(\frac{N}{n+k}+1\right) \times \left(1 - \frac{\sqrt{\left[\sum_{j=1}^{n}\left(tf_{ij} - \overline{tf}\right)^2\right]}}{\frac{k-1}{\overline{tf}}}\right)}{\sqrt{\sum_{i=1}^{N} tf_{ij}^2 \times log^2\left(\frac{N}{n+k}+1\right) \times (1 - \frac{\sqrt{\left[\sum_{j=1}^{n}\left(tf_{ij} - \overline{tf}\right)^2\right]}}{\frac{k-1}{\overline{tf}}})^2}}$$

Wherein, $w_{ij}$ represents feature items weights in file papers j, $tf_{ij}$ stands by times of which feature items appear in the document j, $\mathrm{IDF} = \log n \times \log\left(\frac{N}{n+k}+1\right)$ indicates anti document frequency of feature items, CD represents the value where c is in j.

## IV. EXPERIMENTAL RESULTS AND ANALYSIS

We use Web crawler to get the Finance and economics, military, entertainment and sports in four categories from People's network. The 4 categories is as experiment language bases. Each category were grabbed 1500 news, in total of 6000 documents are as a total corpus , wherein each class takes 1000, altogether 4000 as the training sets, forming classification ware, and finding the appropriate value of K; Each category left over 500, in total of 2000 are as a test set, testing effects of improved TFIDF algorithm. Table 1 shows the distribution of the experimental data in the category training set and test set.

TABLE I. IN THE CATEGORY OF EXPERIMENTAL DATA DISTRIBUTION TRAINING SET AND TEST SET

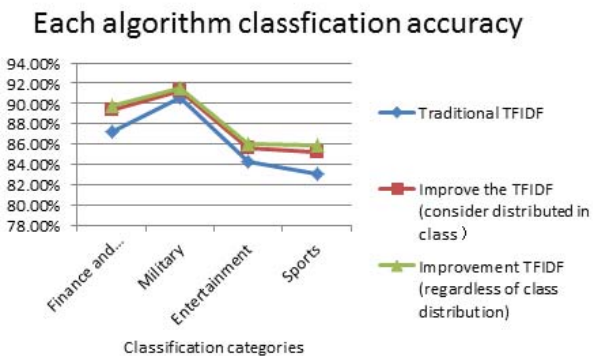
| | *Finance and economics* | *Entertainment* | *Entertainment* | *sports* |
|---|---|---|---|---|
| *Training set* | 1000 | 1000 | 1000 | 1000 |
| *Test Set* | 500 | 500 | 500 | 500 |

We usually use recall and precision to evaluate text classification systems. For a particular category, the recall rate R is defined as: the proportion of number of documents correctly classified documents and the total number of tests, that the probability of being classified correctly identified. Accuracy rate P is defined as: the proportion of number of documents which are correctly classified document and classification recognizes that the classification is to make the right decisions.

Recall and precision rates reflect two different aspects of quality of classification, but both must be considered synthetically. F1 test values considers the influence of the two [13], More higher test value of F1 is, the better results reflect the classification, in this experiment that is reflected in the excellent TFIDF algorithms. F1 test values mathematical formula is as follows:

$$F_1 = \frac{2RP}{R+P}$$

TABLE II. THE TEST VALUES OF DIFFERENT TFIDF ALGORITHM




Average results of each group . From the experimental data in Table II, we can see that, with respect to the value of traditional TDIDF test algorithms, test values of improved TFIDF algorithm has greatly improved. In the test we can see that test values within TFIDF algorithms consider class distribution has more improved than without considering test values within the class of TFIDF, but increase is not very obvious.

While considering TFIDF algorithm performance within the feature words is superior , but it's more complicated algorithm, the time complexity is higher; no thinking about TFIDF algorithm feature words within class distribution is simple, low time complexity. Therefore, we can according to our actual needs and combine with accuracy and time complexity of the algorithm, select the appropriate TFIDF algorithms.

## V. CONCLUSION

This paper treat term weight values as the study contents, summarizing the existing term weight calculation method, describing the advantages and disadvantages of the classic term weight formula, introducing the domestic and foreign classical term weight the formula of research and improvement, analyzing and summarizing their strengths and weaknesses, combining with the impact the distribution of feature words right within the class of heavy text size and characteristics of the word feature that contains the word of weight on text categorization actual impacts. feature words TFIDF algorithm proposed a new calculation method, the traditional end of this article TDIDF. TDIDF algorithm and improved algorithm were carried out experiments to show that

the improved algorithm TFIDF formula is far superior to the traditional TDIDF algorithm, improved real effective sex.

However, due to time constraints and capabilities, as well as a lot of work needs to be improved and deepened. There are several aspects to be improved in future research:

- This article draws on existing segmentation tools and reduction methods, but the effect is not very good, need to be improved in the future scholars.
- The improvement of TFIDF algorithm is only considered the distribution of the feature words in the class and the class, and the correlation between the feature words and the meaning is not considered, In the future work, we need to study and test data and propose a more effective method to improve the existing methods.

## References


[1] JONES K S. A statistical interpretation of term specificity and itsap-plication in retrieval[J]. Journal of Documentation, 1972, 28(1): 11-21.

[2] SALTONG, CLEMENTTY. On the construction of effective vocabularies for information retrieval[C ] //Proceedings of the 1973 Meeting on Programming Languages and Information Retrieval .New York:ACM, 1973: 11.

[3] SALTON G, FOX E A, WUH. Extended Boolean information retrieval[J]. Communications of the ACM, 1983, 26 (11): 1022 -1036.

[4] SALTONG, BUCKLEY C. Term-weighting approaches in automatic text retrieval[J]. Information Processing and Management 1988:513 -523.

[5] BASILI R, MOSCHITTIA, PAZIENZAM. A test classifier based on linguistic processing [C ],Proceedings of IJCAIp99, Machine Learning for Information Filtering, 1999.

[6] HOW B C, NARAYANAN K. An empirical study of feature selection for text categorization based on term weight age[C ] //Proceedings of the 2004 IEEE W/ IC /ACM International Conference onWeb Intelligence. Washington, DC: IEEE Computer Society, 2004: 599 -602.

[7] Yufang Zhang, Shiming Peng, Jia Lu. Improvement and application of text classification TFIDF method [J] Based on. Computer Engineering, 2006, 32 (19): 76-78.

[8] KeliChen, QingZong, XiaWang. Based on the analysis of the text corpus balance classification scale real text [EB / OL]. [2008 -12 -20].

[9] CongyingShi, ZhaojunXu, XiaojiangYang. Review TFIDF algorithm [J]. Computer Applications, 2009,29 (6): 167-170.

[10] Jianhua Su,Guimin Su. Virtual Learning of University Libraries sense of community [J]. Scopus, 2010 (5): 152-154.

[11] JunkaiYi, LikangTian. Algorithm text feature class-based discrimination in the selection [J]. Beijing University of Chemical Technology (Natural Science), 2013,40 (5): 72-75.

[12] LeiHuang, YanpengWu, Qunfeng Zhu. Automatic Keyword Extraction Method Study and Improvement [J]. Computer Science, 2014,6 (6): 204-207.