Study on Term Weight Calculation Based on Information Gain and Entropy

Ying Hong[1, a], Chao Lv[2, b]

[1] Computer Information Center, Beijing Institute of Fashion Technology, Beijing, China

[2] Computer Information Center, Beijing Institute of Fashion Technology, Beijing, China

[a]jsjhy@bift.edu.cn

[b]jsjlc@bift.edu.cn

Keywords: TF-IDF, text classification, term weight calculation, information entropy, information gain

Abstract. This paper first analyzes the advantages and disadvantages of TF-IDF, which is a traditional algorithm of term weight calculation. Then to overcome the disadvantages of the algorithm, this paper proposes a new method of term weight calculation based on information gain and information entropy, which can make the result of the term weight calculation more precise and improve the accuracy of text classification. Finally, the text data sets are downloaded from internet according to the web crawler and 7700 texts are selected randomly as the experimental data sets. The experimental results show that the method proposed in this paper overcomes the disadvantages of the traditional TF-IDF and performs better than the other two in precision, recall, F-measure of the text classification.

1. Introduction

Facing with the increasingly expanding information on the internet, people often feel confused and lost in the vast amount of information resources [1]. How to find the information required accurately and efficiently from these resources have become an important problem for the researchers. The accuracy of term weight calculation has a significant impact on the result of the text classification. This paper improved TF-IDF algorithm based on information entropy and information gain and designed experiments to demonstrate the effect of the improved algorithm.

2. The related technologies of text classification

Text classification divides the text no category labeled into the preset category according to its contents. Documentation set is divided into two parts in text classification process: training set and test set. Training set is used to make classifier learning and test set is used to evaluate the performance of the classifier. Generally, the ratio of training set and test set is 7: 3 [2]. Text classification process generally includes: text preprocessing, text representation, feature item selection, classifier selection and design, classifier evaluation.

2.1 Text preprocessing.

Before text classification, original text data will be preprocessed due to its special nature. Text preprocessing removes the noise factor and adjusts the raw data format for computer processing [3]. Text pretreatment includes many techniques: text segmentation, delete stop words and low-frequency words, delete mark information and so on. In text preprocessing, text segmentation is the most important.

2.2 Chinese word segmentation.

Chinese word segmentation is a unique concept in Chinese text classification [3]. There are no obvious segmentation signs between words in Chinese text. The computer needs to find out automatically the dividing line between words using the Chinese word segmentation. Therefore, the Chinese word segmentation is the foundation of Chinese text classification technology. Now, there are four word segmentation algorithms commonly used: dictionary-based segmentation method,

understand-based segmentation method, statistical-based segmentation method, semantic-based segmentation method.

2.3 Feature selection.

After Chinese segmentation, source text data will be a big set of words which is called the eigenvectors. This set contains up to thousands or even hundreds of thousands of feature words. Most word has little meaning of the text classification [4]. So we must reserve words that best represents the text content for improving the accuracy and efficiency of text classification results. We will construct an evaluation function before feature selection, and then evaluated the feature items in the feature set by the evaluation function. Next, we will screen the assessment results according with certain rules and select the appropriate feature items used as a text representation. The common feature selection methods are: document frequency, mutual information, information gain, z2 statistics, cross entropy, fisher discriminant and other methods.

2.4 Text representation.

After text preprocessing and feature selection, text data is still fragmented and unstructured. The unstructured data can't be learned by text classifier [5]. The most commonly used text representation methods are: Boolean model, probabilistic model, and the vector space model.

2.5 Calculation of **feature weight.**

When using the vector space model (VSM) to represent the text, also need to calculate the weight of feature words [6]. Because different characteristic words has different ability in text classification, the feature words which including more information for text classification should be given a higher weight. On the contrary, the feature words including less information should be given less weight. How to calculate the feature weight has a significant impact on the accuracy of the text classification results. Usual feature weight calculation methods are TF-IDF algorithm, Boolean weights, term Frequency (TF), inverse document frequency (IDF).

2.6 Text classification algorithm.

The design of text classification algorithm is the focus of text classification. Commonly used text classification algorithms can be divided into three broad categories: statistical-based method, neural network method and rule-based method [7].

3. Term weight algorithm

The core idea of term weight calculation is giving different weight to terms which selected by the feature selection algorithm. The purpose is to enhance the ability of feature words in text classification.

3.1 TF-IDF algorithm.

TF-IDF is the most commonly used method. 1973, Salton proposed TF-IDF algorithm [8]. TF-IDF weight calculation formula is shown in equations (1):

$$w_{ik} = tf_{ik}(t_k) \times idf(t_k) = \frac{tf_{ik}(t_k) \times \log(N/n_k + 0.1)}{\sqrt{\sum_{k=1}^{n} (tf_{ik}(t_k))^2 \times \log^2(N/n_k + 0.1)}} \quad (1)$$

Among them, w_{ik} is the weight of term t_k in document d_i. $tf_{ik}(t_k)$ is the frequency that t_k appearing in document d_i. $idf(t_k)$ is inverse document frequency of term t_k. n_k is number of text including term t_k.

In TF-IDF algorithm, when a feature word concentrated in a particular type of text, this feature words has high distinguishing ability for this type.

4. Analysis of TF-IDF algorithm

TF-IDF algorithm takes into account the local and global distribution of the characteristics word, but did not consider the distribution the feature words between classes and within classes [9]. Below these two cases will be analyzed.

4.1 Intra-class distribution.

There are two feature words t1 and t2 in a category c, where t1 is distributed in most of the text and t2 is distributed only in rare text. Ability of the feature words t1 which distributed in the same category more to represent this type of content is strong. It should be given a higher weight. As for the feature word t2 which distributed in only very small part of the text, it does not reflect the content of categories c and contains little classified information. In this category, it should be given lower weights. These contents have not been reflected in the TF-IDF algorithm.

4.2 The distribution between classes.

It supposes the number of text in text set D is N and the category number is p. There are two feature words t1 and t2 which have the same document frequency q. Where t1 is uniformly distributed in the p classes and t2 is distributed in r categories. The remaining p-r categories have no feature words t2. Since t1 is evenly distributed in all types, it carries less classified information and should be given a lower weight. Instead, t2 is only concentrated in some categories, it can be a good representative of the class are distributed. It carries strong classified information and it should be given higher weights in term weight calculation. In accordance with TF-IDF formula, their values of IDF are log (N / q + 0.1). At this time, the measurement of term weight can only be measured by TF which reflects only the frequency of feature words in the text. So TF-IDF algorithm ignores the influence of distribution of inter-class in weight calculation.

The reason of TF-IDF algorithm for causing the above two cases, because TF-IDF algorithm processes the text set as a whole.

5. Improved TF-IDF algorithm based on information entropy and information gain

In order to overcome the inadequacies of TF-IDF algorithm and to improve the accuracy of text classification, this paper presents a characteristic words weight calculation method based on information entropy and information gain named TFIDFIGE. The algorithm not only improves the shortcomings of the traditional TF-IDF algorithm, it also reduces the characteristic dimensions of the text classification.

5.1 Entropy.

The related concepts of entropy are defined as follows:

Definition 1: These are *N* messages with the same probability and the probability of each message is *1 / N*. The amount of information carried by each message is shown as follows:

$$-\log p = \log\left(\frac{1}{N}\right) \tag{1}$$

Definition 2: For a given probability distribution $P = (p_1, p_2,\ldots,p_n)$, then the amount of information carried by this distribution is called entropy of *P*. The formula is as follows:

$$I(P) = -\left(p_1 \times \log_2 p_1 + p_2 \times \log_2 p_2 + \ldots + p_n \times \log_2 p_n\right) = -\sum_{k=1}^{n} p_k \times \log_2 p_k \tag{1}$$

5.2 Information gain.

Information gain mainly describes the difference between information entropy the feature words appearing in the text before and after. Information gain is calculated as follows:

$$IG(t) = H(C) - H(C/t)$$

$$= -\sum_{c\in C} p(c)\log(p(c)) + p(t)\sum_{c\in C} p(c/t)\log(p(c/t)) + p(\bar{t})\sum_{c\in C} p(c/\bar{t})\log(p(c/\bar{t}))$$

$$= \sum_{c\in C}\left(p(c,t)\log\left(\frac{p(c,t)}{p(c)p(t)}\right) + p(c,\bar{t})\log\left(\frac{p(c,\bar{t})}{p(c)p(\bar{t})}\right)\right) \quad (1)$$

Among them, *IG(t)* is the information gain of term *t*. *c* represents text variable, *C* represents a text set and $C = (c_1, c_2,\ldots,c_n)$. *H (C)* is the entropy that the probability of a random text belongs to a category before in the absence of characteristic words *t*. *H (C/t)* is the entropy that the probability of a random text belongs to a category after obtaining feature word *t*.

5.3 Improved TF-IDF algorithm Based on information entropy and information gain.

This paper presents a new method named TFIDFIGE to calculate the weight based on information gain and information entropy. The formula is as follows:

$$W_{ik} = tf_{ik}(d_i) \times idf(t_k) \times IG(C,t_k) \times E_{ic}(t_k) \quad (1)$$

$$IG(t_k) = \sum_{c\in C}\left(p(c,t_k)\log\left(\frac{p(c,t_k)}{p(c)p(t_k)}\right) + p(c,\bar{t}_k)\log\left(\frac{p(c,\bar{t}_k)}{p(c)p(\bar{t}_k)}\right)\right) \quad (1)$$

$$E_{ic}(t_k) = \sum_{j=1}^{n}\frac{tf(t_k,d_j)}{tf(t_k,C_k)}\log_2\frac{tf(t_k,d_j)}{tf(t_k,C_k)} \quad (1)$$

Among them, W_{ik} is the weight of t_k in text d_i. $f_{ik}(d_i)$ is the frequency of t_k in text d_i. $idf(t_k)$ is inverse document frequency of t_k. $IG(C,t_k)$ is the information gain value of t_k. $E_{ic}(t_k)$ is weighting factor of distribution information in class.

6. Method validation experiments

In this paper, we use the web crawler to download text from the internet. We randomly selected 7700 as the experimental data set. It is divided into seven categories: culture, entertainment, history, military, reading, social and legal. KNN classification is adopted in the experiment. The following Table I indicates KNN classifiers experimental results when k=100.

Table 1 experimental results

Categories	TF-IDF			TFIDFIGE		
	Average precision (%)	***Average recall (%)***	*F1* ***Value (%)***	***Average precision (%)***	***Average recall (%)***	*F1* ***Value (%)***
culture	75	57.3	64.9	80.6	65	71.8
entertainment	87	97.3	92.1	93	92.5	92.7
history	70.1	76	73	76.5	84	79
military	69	93.2	79.2	76.8	92	83
reading	87	78	74.2	88.9	98.2	95.1
social	91.6	90	91.5	94.5	96.2	95.1
legal	88.7	90.3	90.2	92	95.4	94.2
mean value	81.2	83.2	80.7	86.0	89.0	87.3

As we can see from Table I, the classification accuracy has been increased obviously when we used the improved TF-IDF algorithm based on information entropy and information gain.

7. Summary

This paper introduced the Chinese word segmentation process and studied the feature selection algorithm. It improved TF-IDF algorithm based on information entropy and information gain and designed experiments to demonstrate the effect of the improved algorithm. The results show that the improved algorithm improves the accuracy of classification obviously.

Acknowledgements

This work was financially supported by the Foundation for Beijing teacher team construction-youth with outstanding ability project (No.YETP1414), the twelfth five-year plan important subject of Beijing education science research (No. AJA11174) and the scientific research program of Beijing institute of fashion technology (No.2012A-17).

References

[1] Zhongbao Liu: Construction of User Interest Model in Personalized Search Engine. Computer Systems & Applications, 2012, 21(11): 1-6(In Chinese)

[2] Lin Chen, Jian Wang: Comparison and research on algorithms of three Chinese text classification. Computer and Modernization, 2012, 2: 1-4(In Chinese)

[3] Zongren Zhang, Tianqi Yang: Personalized Meta Search Engine Based on Subject Tree. Computer Engineering and Design, 2011, 32(1): 149-152(In Chinese)

[4] Zhe Wang: Study and comparation on feature selection method in Chinese text categorization. Journal of Computer Applications, 2011, 19: 18-20(In Chinese)

[5] Anjiang Lu, Xuhui Dong: Research and Design of Personalized Meta-search Engine Model. Computer and Modernization, 2011(1): 139-141(In Chinese)

[6] Xiaoli Li, Zhenlong Du: Investigation on Personalized Search Engine Based on Lucence. Computer Engineering, 2010, 36(19): 258-260(In Chinese)

[7] Zhe Zhao, Yang Xiang, Jisheng Wang: Text classification based on parallel computing. Journal of Computer Applications, 2013, 33(S2): 60-62, 66(In Chinese)

[8] Zhi Chen, Yanyu Qian: Study Personalized Search Engine Based on User.s Interest. Journal of Hefei Normal University, 2010, 28(3): 79-81 (In Chinese)

[9] Wenli Gu, Wei Chen, Jiao Chen, Xiaoye Lu: Improved PageRank Algorithm. Computer Systems and Applications, 2012, 21(2): 214-217(In Chinese)