Advanced Materials Research Vols. 403-408 (2012) pp 1791-1794
Online available since 2011/Nov/29 at www.scientific.net

doi:10.4028/www.scientific.net/AMR.403-408.1791

Improved Algorithm Based on TFIDF in Text Classification

Hao Jiang[1, a], Wenqiang Li[2,b]

[1, 2] School of Computer Science & Engineering ,Southeast University Nanjing ,China

[a] hjiang@seu.edu.cn, [b] rein07@163.com

Keywords: TFIDF; NTFIDF; text classification; KNN

Abstract. Traditional feature weighting algorithm TFIDF doesn't take some other factors which impact the feature weight into consideration, so this paper discusses the factors in details and proposes a new feature weighting algorithm called NTFIDF combined with these factors and TFIDF. Experiment on the KNN classifier shows that NTFIDF is better than TFIDF in text classification.

Introduction

With the development and popularization of networks and computers, more and more information come into people's perspective. In order to get the information, what people need to do is powering the computer on. However, it is not so easy to get what you are interested in from the massive data. So it makes information classification much more important, among which text classification is the core because of much of the information in stored in text. The key in text classification is how to select effective features as the effective features of category and how to compute the weight of the effective features[1,2]. The traditional feature weighting algorithm is TFIDF which lacks of some other factors which impacts the feature weight, so this paper considers those factors in terms of word probability, document probability and document frequency, combines these factors with TFIDF to form a new feature weighting algorithm called NTFIDF. Finally, experiment on the KNN classifier shows that NTFIDF is better than TFIDF in text classification.

1 TFIDF algorithm

1.1 TFIDF

Term Frequency[3]: The number of occurrences of the feature appears in the document. Simply use the TF often leads to the following questions:① A large number of function words which hardly do contribute to text classification and appear in the document such as interjection, preposition and conjunction etc will have a negative impact on classification if they are selected as the features;② Selected feature words which are good or not depend on whether they are on behalf of the property of category and document, a feature word with a high value of TF is hard to classify which category it presents for provided that the value is all high in all categories.

Inverse document frequency(IDF)[3]: the quantitative of feature distribution in document set, it can weak the importance of high-frequency features which appear in many documents, meanwhile enhance the importance of low-frequency features which appear in very few documents.

The computational method of TFIDF is :

$$\mathrm{IDF(t)}=\log(\frac{N}{n}) \tag{1}$$

In the above, N presents the number of documents in the document set and n presents the number of documents which the feature t appears in.

An effective feature is the one which not only presents the contents of the category it belongs to, but also makes the category distinct with the other category. So the TFIDF is born as follows:

$$Weight_{TFIDF} = TF(t) \times IDF(t) \tag{2}$$

1.2 Shortage of TFIDF

The main idea[4] of TFIDF is that the feature is good for text classification only if the value of TF of the feature t appeared in a document is high and is low in the others. The main idea of IDF is that the feature is good for text classification only if the value of IDF of the feature t appeared in the documents is high which means the number of documents which the feature t appeared in is small. [4] suggests that TFIDF is not good enough whether within the category or between the categories because of lack of the other factors which impact the feature weight.

Between-categories: if the number of documents which contain feature t in category c is m and the number of documents which contain feature t in other categories is k, so the total number of documents which contain feature is n=m+k . n increases as m increases, so the value of IDF is low which means the capability of feature t used in text classification decreases. In fact, when m is higher, it means the feature t appeared in category c is good to present the category, so the value of the feature t should be higher. This is one side of the shortage. On the other side, if the n is low, which means the number of documents containing the feature t is small , and the documents are distributed evenly in the document sets, the feature t should not be taken as a good feature for the category and should be given a lower value, but in fact, the value of IDF is higher. What leads to these problems is that TFIDF is considered in term of a whole of documents, not of the factors within the category.

Within-category: in the features appeared in a categories, the value of those distributed evenly should be higher than the value of those distributed unevenly. Because of the features appeared in only a few documents, it can be think of a special circumstances and not good enough to present the whole category. TFIDF can not handle the situation gracefully.

2 Improved TFIDF algorithm

Considering the shortage of TFIDF, we analyze the factors that affect the feature weight carefully and try to use these factors to improve TFIDF algorithm.

Considering the first one of the shortages in between-categories, the reason is that one factor called document frequency(NT) are not included in TFIDF. NT is the number of documents which contain feature t in category c. If it is added into TFIDF algorithm, the feature weight will increase or decrease with NT, so it resolve the first problem effectively.

Considering the second one of the shortages in between-categories, the reason is that document Probability are not included in TFIDF. document probability is the probability of documents which contain the feature t in category c. It can effectively resolve the problem of even distribution of documents containing feature t in category set.

Finally considering the last one of the shortages in within-category, the reason is that word Probability is not included in TFIDF. It can effectively resolve the problem of uneven distribution of features appeared in category c. If the average probability of the feature is high, It can be taken as a effective feature for the category.

At last, we combine these factors with TFIDF to form a new feature weighting algorithm called NTFIDF. Formula is as follows:

$$Weight_{NTFIDF} = TF(t) \times IDF(t) \times NT(t) \times \frac{NT(t)}{DF(t)} \times \frac{W_C(t)}{W(t)} \tag{3}$$

In the above, $NT(t)$ presents the number of documents containing feature t in category c. $DF(t)$ presents the number of documents containing feature t in category set. $\frac{NT(t)}{DF(t)}$ presents the probability of documents containing feature t. $W_C(t)$ presents the total number of words appeared in category c. $W(t)$ presents the total number of words appeared in category set. $\frac{W_C(t)}{W(t)}$ presents the word probability of feature t in category c.

3 KNN algorithm

KNN(K-nearest)[8] classification algorithm is a method based on similarity between vectors. It's main idea is that: for an unspecified text Z, the system finds K nearest neighbors for Z, the categories the K texts belong to are the candidate categories for Z. Then the category which has a largest number of texts in K texts is the category of Z. Computing the distance between test samples and training samples is as follows:

$$Sim(d_i, d_j) = \frac{\sum_{k=1}^{N} w_{ik} \times w_{jk}}{\sqrt{\sum_{k=1}^{N} w_{ik}^2} \sqrt{\sum_{k=1}^{N} w_{jk}^2}} \tag{4}$$

KNN is a lazy classification algorithm[9]. The needed computation are delayed until classification time. A large number of training vectors are stored in the KNN classifier. When classifying , the distance between the unspecified text vector and the training vector. Because it has a better classification results, so it has a wide range of application in text classification. For high-dimensional text vector and larger sample set, time and space complexity is high, the cost is O(m*n), n is the dimensions of VSM, m is the scale of sample set. Study shows that KNN requires a large number of training samples to get a more precise classification.

Its advantages are mainly: one is that it can play an important role under unbalanced sample circumstances, because KNN only has relation with a few similar samples; second is that KNN can make up greatly the errors caused by feature selection; the third is that KNN is a better-effective algorithm.

4 Experimental results and analysis

4.1 Experiment description

Text classification using KNN algorithm to evaluate the merits of feature weights. the experimental data is provided from the corpus of Fudan University among which there are 10 categories in test and training samples. Its details shown in Table 1.

Table1 Experimental Data

Type	category									
	Environment	Computer	Traffic	Education	Economy	Military	Sports	Medicine	Arts	Politics
Taining	134	134	143	147	217	116	301	136	166	338
test	67	66	71	73	108	83	149	68	82	167

4.2 Analysis

This paper evaluates the experimental results in terms of the average macro and micro-average of recall, check rates, F1 value. These definitions are in document [5].

From table 2, it shows that improved algorithm NTFIDF is better than traditional algorithm TFIDF, every type of average macro and micro-average of NTFIDF is better than that of TFIDF.

Table 2 Experiment Results

algorithm	Indicators					
	recall		check rates		F_1	
	average macro /%	micro-average /%	average macro /%	micro-average /%	average macro /%	micro-average /%
TFIDF	85.765	90.779	88.116	88.116	88.116	88.116
NTFIDF	86.752	90.890	88.865	88.865	88.865	88.865

5 Conclusion

This paper describes the shortage of TFIDF in calculation of the feature weight, then analyze the other factors which affect the feature weight, finally combine these factors with TFIDF to form a new algorithm called NTFIDF. Experiment on the KNN classifier shows that NTFIDF is better than TFIDF in text classification.

References

[1] Huanling Tang，Jiantao Sun，Yuchang Lu．A Weight Adjustment Technique With Feature Weight Function Named TEF-WA In Text Categorization [J]．Journal of Computer Research and Development，2005，42(1)：47.53.

[2] Ning Zhang，Ziyan Jia，Zhongzhi Shi．Text Categorization With KNN Algorithm [J]．Computer Engineering，2005，31(8)：171.172.

[3] Rongjun feng. Improvement And Application Of Document Frequency-Based Feature Extraction Algorithm [D]. Nanjing University of Posts and Telecommunications ，2005

[4] Jia Lv. Improved Feature Selection Algorithm Based On Variance In Text Categorization [J]. Computer Engineering And Design，2007,28(24)

[5] Jianhui Wang，Hongwei Wang，Zhan Shen．A Simple and Efficient Algorithm to Classify a Large Scale of Texts [J]．Journal Of Computer Research And Development. 2005，42(1)：85-93.

[6] Xianqun Tong，Zhongmei Zhou. Enhancement Of K-Nearest Neighbor Algorithm Based On Information Entropy Of Attribute Value [J]. Computer Engineering And Applications, 2010，46(3).

[7] Guang Rong. Study of Chinese Text Classification Method [D]. Shandong University，2009.

[8] Dongmei Liu. Automatic Classification Research On Html Document And Implentation Of The Tool[D]. Inner Mongolia University,2006.

[9] Xiaohua Zhao. Research Of Weight Algorithm In Text Cassification [D]. Taiyua University of Technology,2010.

MEMS, NANO and Smart Systems
10.4028/www.scientific.net/AMR.403-408

Improved Algorithm Based on TFIDF in Text Classification
10.4028/www.scientific.net/AMR.403-408.1791