Research on Text Categorization of KNN Based on K-means for Class Imbalanced Problem

Wang Yu
School Of Computer Science and Technology
Tianjin University
Tianjin, China
csyuwang@sina.com

Xu Linying
School Of Computer Science and Technology
Tianjin University
Tianjin, China
linyingxu@tju.edu.cn

***Abstract*—With the rapid development of Web and the rapid expansion of text information, how to effectively organize and manage these information is a great challenge for the current information science. Text automatic classification technology can effectively organize a large number of texts and help people to improve the efficiency of information retrieval. It has become one of the most important research directions in the field of information processing. There are many mature methods of text classification, where K-Nearest Neighbor algorithm has good accuracy, it is suitable for multiple classification problems and has been widely used in the field of document classification. However, when dealing with the training set with class imbalanced problem, the classification results tend to be biased towards majority class, so that the accuracy of the classifier is greatly reduced. In order to solve this problem, two strategies that construction of samples based on clustering and weighted KNN based on sample density are proposed in this paper to improve the traditional KNN algorithm. Four datasets which have different class imbalanced rates are extracted from the entire corpus, and we use classic KNN, NWKNN and Kmeans-KNN algorithm to perform cross validation on each dataset. The results show that compared with the traditional KNN algorithm and NWKNN algorithm, the proposed method can effectively improve the classification accuracy and G-mean value, and has better stability under the class imbalanced problem.**

Keywords—Text categorization; K-Nearest Neighbor; K-means; Class imbalanced problem;

I. INTRODUCTION

Text information is the main carrier of Web resources, according to the text information and predefined topic categories, the process of automatically classifying the text in the document collection into a certain category based on certain rules is called text categorization. Early text categorization is mainly finished through a series of rules that manually defined, manually defining classification rules is very time consuming, and it is necessary to have a sufficient understanding of the knowledge in a particular field to give a suitable and effective rule. With the rapid development of Internet, the emergence of a large number of electronic texts and the rise of machine learning, large scale automatic text classification technology is receiving more and more attention. Therefore, it has become one of the most important research directions in the field of information processing [1]. Text categorization based on machine learning can help people to organize the text better and collect text information, text classification techniques can be used to organize a large number of texts, which can help people to improve the efficiency of information retrieval. The text classification system firstly uses some kind of classification algorithm to train the text set which has already been classified to get a text classifier, which can be used to classify the text automatically. Decision Tree, Naive Bayes, Neural Network, Support Vector Machine and K-Nearest Neighbor are common algorithms used in text classification. A large number of experimental results show that the classification accuracy of these methods is comparable to the results of manual classification of experts, and the learning process does not require expert intervention, it greatly improves the classification efficiency and can be applied to any field of learning.

With the continuous development of machine learning, artificial intelligence and other disciplines, many mature text categorization methods are formed. K-Nearest Neighbor classification algorithm is simple and effective, and is suitable for multi-classification problems, thus widely used in the field of text classification. The traditional KNN classification algorithm uses the class of the most similar K samples to determine the classification of the samples to be classified. This leads to the performance of classification is susceptible to the influence of the sample distribution. When the sample set has a class imbalanced problem, namely the sample size of some class is very large and the sample size of the other classes is relatively small, the k nearest neighbors which classifier selects are mainly from the classes that have more samples. As a result, a new test sample is tends to be classified as the class which has more training samples described by [4][5][6]. Many improved algorithms are proposed in [9][10][11][12]. To solve this problem, an improved KNN algorithm based on K-means clustering is proposed in this paper, firstly, according to the clustering results of K-means, some new training samples that belong to small classes are constructed, then in the KNN algorithm, it gives the training samples in small classes a larger weight factor. The improved algorithm has higher classification accuracy and stability, it is suitable for the dataset with all kinds of distribution, especially for the dataset with class imbalanced problem.

The rest of this paper is organized as follows: section II introduces the K-Nearest Neighbor algorithm, K-means algorithm, and the effect of class imbalanced problem on KNN.

978-1-5090-1195-7/16 $31.00 © 2016 IEEE
DOI 10.1109/IMCCC.2016.61

The Kmeans-KNN algorithm is descried in detail in Section III. In the section IV, the text classification experiment is designed on the Chinese news dataset, and the classification results of classical KNN, NWKNN and Kmeans-KNN are analyzed and compared. Finally, the conclusion is drawn in section V.

II. Related Work

A. Classical KNN Algorithm

In text classification, K-Nearest Neighbor algorithm is a kind of machine learning method based on the instance, it does not build a model until the classification of new instances. This also means that the KNN algorithm does not have a real sense of the training stage. The basic idea of the algorithm is: given a new document, it calculates the similarities between the feature vector of this document and each of document vectors in the training set, then it gets K documents which has the highest similarity with the new document, at last, it determines the class of the new document according to the classes of these K documents. The specific algorithm steps are as follows:

1) Calculate the similarities between the document to be classified and each of documents in the training set. The computation formula is defined as:

$$\mathrm{Sim}(d_i, d_j) = \frac{\sum_{k=1}^{M} W_{ik} \times W_{jk}}{\sqrt{(\sum_{k=1}^{M} W_{ik}^2)(\sum_{k=1}^{M} W_{jk}^2)}} \quad (1)$$

Where $d_i = (W_{i1}, W_{i2}, \dots, W_{iM})^T$ and $d_j = (W_{j1}, W_{j2}, \dots, W_{jM})^T$ are the feature vector of the document.

2) Select K documents with the highest similarities according to the similarities between the document to be classified and all the training documents, and calculate the weight of each class according to these K neighbors. The computation formula is defined as:

$$p(x, C_i) = \sum_{d_j \in \mathcal{D}_{KNN}} Sim(x, d_j) y(d_j, C_i) \quad (2)$$

Where x is the feature vector of the document to be classified, C_i is the class of the document i, $\mathcal{D}_{KNN}$ is the collection of the neighbor doucments of x, $Sim(x, d_j)$ is the similarity between the document to be classified and the j neighbor document, $y(d_j, C_i)$ is an indication function which means that if d_j belongs to class C_i, then the function value is 1, otherwise 0.

3) Compare the weights of different classes and assign the document to the class with largest weight. Class decision function is defined as:

$$C_{belong} = arg \ \ max_{C_i}(p(x, C_i)) \quad (3)$$

B. Effect of Class Imbalanced Problem on KNN Classifier

In many machine learning tasks, the training set may have class imbalanced problem that the number of samples under some classes is far more than other classes. The classification effect of KNN is easy to be affected by the data distribution, given an imbalanced data distribution, the samples under majority classes are easy to be selected, so that they account for the majority of nearest neighbors selected by the KNN classifier.

At present, there are a lot of research on how to improve the accuracy of the classification results of the KNN algorithm. Common methods include the combination algorithm and the improvement of the classical KNN algorithm. The combination algorithms proposed in [2][3] are AdaCost and RareBoost algorithm respectively, the basic idea is to use the integrated approach to train many different classifiers, then combine with these classifiers into one. It alleviates the imbalanced problem, but training multiple classifiers greatly increases time overhead. The common method of improved KNN is to punish the model, it will increase the weight of samples under minority class while reducing the weight of samples under majority class, which makes the classifier focus on minor samples. Based on this idea, literature [4] put forward NWKNN algorithm based on the weight function which gives samples different coefficients according to their class imbalanced degrees. In addition, a variety of weight factors based on the data distribution are proposed in [5][6][7][8], But these methods only increase the weight coefficient of samples under minority class to improve the accuracy, overfitting occurs easily due to very few samples. Therefore, the improved KNN classifier based on K-means is proposed in this paper. In order to avoid the overfitting, it generates a certain number of samples under minority class according to the center points computed by K-means and calculates the weight factor of each sample based on the Gaussian function.

III. Improved KNN Text Classifier Based on K-means

The dataset can be more balanced by resampling and thus the classification performance will be improved. For samples under minority classes, it is easy to copy a part of the original samples and add them to the dataset, however, this method does not increase the amount of information in the training set, so the performance of the classifier will not be significantly improved. Therefore, we use a sampling method based on K-means to improve the KNN classifier, so that it can still have a good classification performance in case of class imbalanced problem.

A. Construction of Samples Based on Clustering

In the KNN text classifier, for a document to be classified, the central points are better than the edge points as far as the role of determining classes is concerned. That is, the classification performance is more dependent on the representative samples in each class. However, for a certain class, using the geometric mean of all the samples as a central point may not be well applied to text classification. In fact, the theme of different documents in a particular class is different, so there are many different themes which are in different geometric positions under the class. Therefore, the geometric mean of all samples cannot correctly reflect the samples under different themes, and it will ignore the samples on the edge of data space.

Based on the above considerations, K-means clustering is carried out in the classes with few samples and generates K cluster centers which represent different themes, it is assumed that the data of these clusters is satisfied with the multivariate Gaussian distribution, whose mean is the cluster center point, and the covariance matrix is the unit matrix. Then for each cluster center, a large number of sample points are generated according to the corresponding multivariate Gaussian distribution, and select a certain number of sample points to add

to the original dataset according to the weight measured by the similarity of the new generated sample points and the cluster centers.

B. Weighted KNN Based on Sample Density

In this paper, we add the weight factors of different classes based on Gaussian function. This makes weight factor of the sample under majority class is small so that the weight factor λ_i of the majority class is reduced. The weight factor λ_i of the class i is defined as:

$$\lambda_i = \frac{e^{-(\frac{n_i}{N})^2/2\sigma^2}}{\sqrt{\sum_{i=1}^{r}(e^{-(\frac{n_i}{N})^2/2\sigma^2})^2}} \tag{4}$$

Where n_i is the number of samples under class C_i, N is the total number of training samples, σ is the width of the Gaussian function, the greater n_i is, the smaller λ_i is.

The formula of the weight under each class is defined as:

$$p(x, C_i) = \sum_{d_j \in \mathcal{D}_{KNN}} \lambda_i \times Sim(x, d_j) y(d_j, C_i) \tag{5}$$

he equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

C. Weighted KNN Text Classifier Based on Clustering

In summary, the improved KNN text classifier is constructed as follows:

1) Cut training documents into words and use the TF-IDF algorithm to calculate the weight of each word, then the document is expressed as an n-dimensional vector $(x_1, x_2, \dots, x_n)$, where x_i represents the weight of the word i.

2) K-means clustering algorithm is used to divide the training samples under each class into K clusters, and the similarity of two documents is measured by the cosine similarity of the word vector.

3) For the class with few samples, construct new samples based on the cluster center and add to the original dataset.

4) Find out its K nearest neighbors according to the similarities between the document to be classified and each document in the training set.

5) Compute and compare the weights of different classes and assign the document to the class with largest weight.

IV. Experiment and Result Analysis

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

A. Dataset and Evaluation Criteria

In this paper, we use the text classification corpus in Sogou Laboratory as the experimental data, the content comes from the news corpus of Sohu, and it consists of 17910 documents and comes from 9 categories, including finance, IT, health, sports, tourism, education, recruitment, culture and military. We extract 4 datasets from the entire corpus which have different class imbalanced rates. The data distributions of them are shown in Table I.

TABLE I. The Data Distributions of Different Dataset

Data set	Total	Finance	IT	Health	Sports	Tourism	Education	Recruitment	Culture	Military	Imbalanced Rate
D1	2215	66	115	55	15	965	300	315	100	20	0.015
D2	2218	56	532	65	300	50	300	495	382	38	0.071
D3	2033	95	296	170	148	257	402	280	90	295	0.224
D4	2205	295	295	295	295	295	295	295	295	295	1.000

The class imbalanced rate of each dataset is measured by the ratio of the number of samples in minimal class and the number of samples in maximum class. The smaller the ratio, the greater the degree of class imbalanced of the dataset.

90% accuracy is obtained when the classification is performed on the imbalanced dataset. But further analysis find that 90% of the samples are in the same class, and the all the data are classified into that class. In this case, it is clear that the classifier is invalid. And this kind of invalidation is caused by the class imbalanced problem. As can be seen, when the class is imbalanced, using classification accuracy to measure the quality of the classifier cannot reflect the effect of the classifier. Therefore, in this paper, we use precision, recall, F1 value and G-mean as the evaluation standard of the classifier. These are based on the confusion matrix, as shown in Table II.

TABLE II. Confusion Matrix

Class	Prediction positive class	Prediction negative class
Actual positive class	TP	FN
Actual negative class	FP	TN

The formula of precision, recall, and the F1 value are as follows:

$$\text{precision} = \frac{TP}{TP + FP} \tag{6}$$

$$\text{recall} = \frac{TP}{TP + FN} \tag{7}$$

$$\text{F1} = \frac{precision \times recall \times 2}{precision + recall} \tag{8}$$

$$\text{G-mean} = \sqrt[n]{\prod_{i=1}^{n} Accuracy_i} \tag{9}$$

Where $Accuracy_i$ is the classification accuracy of the class i, n is the total number of classes, G-mean evaluates the overall classification performance of the classifier.

B. Experimental Design and Parameter Setting

In this paper, we use the classical KNN, NWKNN[4] and Kmeans-KNN three different algorithms to experiment and compare the classification results. All classifiers set the nearest neighbor number K=21. The exponent parameters of NWKNN algorithm are set to 3 and 5 respectively and the corresponding

classifiers are denoted as NWKNN3 and NWKNN5. The number of cluster centers in Kmeans-KNN algorithm is set to 20.

We use KNN, NWKNN2, NWKNN4 and Kmeans-KNN these four kinds of classifiers to take the experiment on the above four different datasets with test mode of 4-fold cross-validation. In addition, Kmeans-KNN classifier will generate N sample points for each cluster center, therefore, we also test on the D2 dataset by using Kmeans-KNN classifier for different factor N.

C. *Experimental Results and Analysis*

Experimental results of the D1 dataset of four classifiers are shown in Fig. 1. As can be seen from the results, the classification accuracy of the classical KNN classifier under health, sports, military and other minority classes is very low, respectively, 56.7%, 35% and 16.5%. The classification accuracy of NWKNN3 classifier is increased to 70.3%, 54.2% and 67% respectively, while NWKNN5 is raised to 68.7%, 59.2% and 67% respectively. Compared with the above methods, the Kmeans-KNN classifier has the most obvious improvement, the classification accuracy is 73.4%, 85% and 90.2% under health, sports and military. In addition, compared to the classic KNN classifier, the F1 value of Kmeans-KNN classifier under health, sports and military classes is increased by 6.7%, 24.8% and 46.8%, its G-mean value is increased by 0.16. In a word, the classification accuracy of Kmeans-KNN classifier under minority classes and the overall G-mean value is higher than that of the classical KNN classifier and the NWKNN classifier.

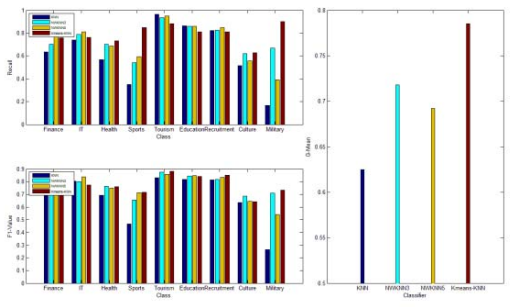

Fig. 1. Experimental results of D1 dataset

Experimental results of D2 dataset are shown in Fig. 2.

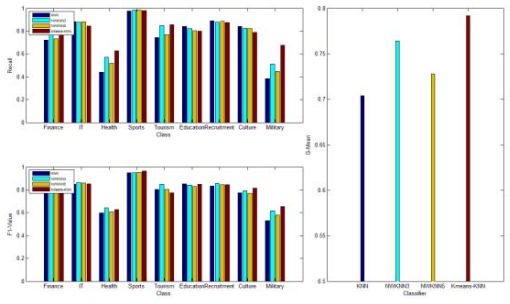

Fig. 2. Experimental results of D2 dataset

As shown in Fig. 2, the classification accuracy of the classical KNN classifier under finance, health, tourism and military is relatively low. Although two kinds of NWKNN classifier also improve the classification accuracy under minority classes, it is not as good as the Kmeans-KNN classifier, and the G-mean value of the improved KNN classifier proposed in this paper is the highest, so its classification performance is better than the first two classifiers.

Experimental results of the dataset D3 are shown in Fig. 3. It can be seen that the classification accuracy of Kmeans-KNN classifier is higher than the classical KNN classifier, but only increased by 13.3% and 10.9% respectively. Its G-mean value is increased by only 0.01.

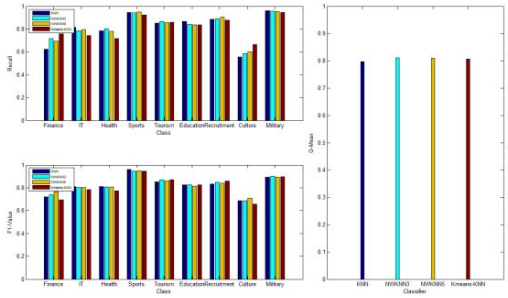

Fig. 3. Experimental results of the D3 dataset

Experimental results on the D4 dataset are shown in Fig. 4. It can be seen that each classifier has a higher classification accuracy under each class, and it has no obvious difference. Under the completely balanced dataset, the improved KNN classifier and the classical KNN classifier have little difference in the classification results.

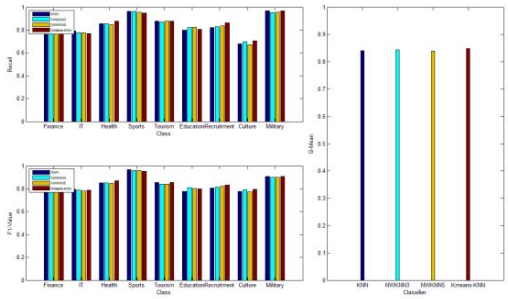

Fig. 4. Experimental results of D4 dataset

According to the experiments on four datasets with different imbalanced degrees, the G-mean values of four classifiers are obtained for each dataset. The results are shown in Fig. 5. As can be seen, the greater the imbalanced degree is, the greater the increase in G-mean value of the improved KNN classifier is. Compared to the traditional KNN classifier, the improved classifier has better classification performance, but with the decrease of the class imbalanced degree, the increase in G-mean value is getting lower and lower. When the data distribution tends to be balanced, there is not much difference between the classification performance of traditional KNN classifier and the

improved KNN classifier. Therefore, Kmeans-KNN has a good classification accuracy under each class, and it is able to adapt to the different distribution of the dataset, its stability is better than other classifiers.

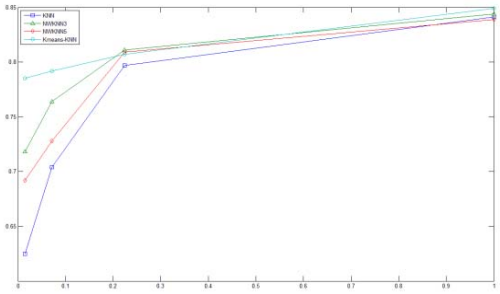

Fig. 5. The variation of G-mean value of different classifiers

We test on the D2 dataset by using Kmeans-KNN classifier for different factor N. With the increase of N, the classification accuracy under finance, health, tourism, military and other minority classes shows an upward trend, while the classification accuracy under majority classes shows a downward trend. The trend of classification accuracy under partial classes and the G-mean value with factor N are shown in Fig. 6. As can be seen from the figure, the G-mean value increases at first and then decreases, and it reaches the maximum when N=8.

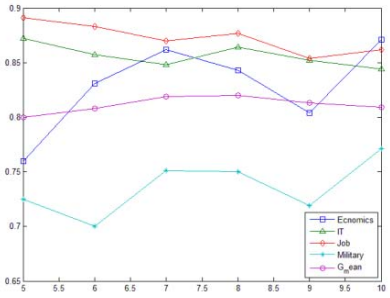

Fig. 6. The trend of classification accuracy and the G-mean value

V. CONCLUSION

In this paper, an improved KNN classifier based on K-Means clustering is applied to Chinese text classification, which makes the classification results more accurate and stable. In order to overcome the defect that the traditional KNN algorithm tends to the majority class, the dataset is resampled, firstly, we calculate the K cluster centers of minority class by K-means, and it is assumed that the sample data under these clusters satisfy the multivariate Gaussian distribution, then for each cluster center, a large number of sample points are generated according to the corresponding multivariate Gaussian distribution, and select a certain number of sample points to add to the original dataset according to the weight measured by the similarity of the new generated sample points and the cluster centers. In addition, the Kmeans-KNN algorithm also uses the Gaussian function as the weight function of the samples, so that it can improve the weight factor of the samples in minority class. The above two methods are combined to solve class imbalanced problem effectively.

Four different types of experimental datasets are chosen in this paper. Each dataset has different degree of imbalance. For each dataset, it is tested respectively by classic KNN algorithm, NWKNN3, NWKNN5 and Kmeans-KNN algorithm with test mode of 4-fold cross-validation. In particular, we divide the data into four groups and do the experiment four times, each time one group of data are used as the test set, the other three groups are used as the training set, the average of these four results is taken as the final test result. The comparative analysis of experimental results shows that for an imbalanced dataset, compared to the other three algorithms, Kmeans-KNN algorithm has a higher classification accuracy and g-mean values, so the algorithm can more effectively solve class imbalanced problem and it is more stable because the classification results are basically consistent under the different distribution of data.

REFERENCES

[1] Jin-Shu S U, Zhang B F, Xin X U. Advances in Machine Learning Based Text Categorization[J]. Journal of Software, 2006, 17(9).

[2] Zhang J. AdaCost: Misclassification Cost-sensitive Boosting[J]. 1999.

[3] Irawan A C. Analisis Algoritma RareBoost-1 Dalam Kasus Imbalance Class[J]. 2007.

[4] Tan S. Neighbor-weighted K-nearest neighbor for unbalanced text corpus[J]. Expert Systems with Applications, 2005, 28(4):667-671.

[5] Hao X, Tao X, Xu H, et al. A Strategy to Class Imbalance Problem for kNN Text Classifier[J]. Journal of Computer Research & Development, 2009, 46(1):52-61.

[6] Liu H F, Pang X M, Zhang X R. An Improved Density-Based KNN Algorithm under Clustering[J]. Microelectronics & Computer, 2011, 28(7):125-124.

[7] Liu H F, Chen Q, Liu S S, et al. An Improved KNN Text Categorization Method Based on Data Uneven[J]. Microelectronics & Computer, 2010, 27(3):51-53.

[8] Liu H F, Yao Z Q, Zhan S U, et al. A Clustering-Based Method for Reducing the Amount of Sample in KNN Text Categorization on the Category Deflection[J]. Microelectronics & Computer, 2012, 29(5):24-28.

[9] Guo, Hongyu, Viktor, Herna L. Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach[J]. ACM SIGKDD Explorations Newsletter, 2004, 6(1):30-39.

[10] Wu G, Chang E Y. Class-boundary alignment for imbalanced dataset learning[J]. Icml Workshop on Learning from Imbalanced Data Sets, 2010:49--56.

[11] Wang C X, Pan Z M, Chun-Sen M A, et al. Classification for Imbalanced Dataset of Improved Weighted KNN Algorithm[J]. Computer Engineering, 2012.

[12] Batista G E A P A, Prati R C, Monard M C. A study of the behavior of several methods for balancing machine learning training data[J]. Acm Sigkdd Explorations Newsletter, 2004, 6(1):20-29.