International Journal of Hybrid Information Technology
Vol.8, No.3 (2015), pp.63-72
http://dx.doi.org/10.14257/ijhit.2015.8.3.07

The KNN based Uyghur Text Classification and its Performance Analysis

Palidan Tuerxun[1, 2], Fang Dingyi[1] and Askar Hamdulla[2]

[1] School of information and technology, Northwestern University, Xi'an, China
[2]School of Software, Xinjiang University, Urumqi, Xinjiang, China
askarhamdulla@gmail.com

Abstract

This paper takes the automatic classification of the large-scale Uyghur text collected from the network as research background, designed the functional block structure of the Uyghur text classification system, and chose the KNN algorithm as the classification engine, and programmed the classification system using C sharp. In the preprocessing part, combining with the Uyghur language's lexical characteristics, we introduced the stem extraction method into the procedure, and then have greatly reduced the whole feature dimensions. the classification experimental results on the basis of large-scale text corpus includes more than 3000 documents which are belongs to different 10 categories are given, and the results of the classification experiments for the different number of features selected by using x2 statistical method are also given. The results show that only 3% to 5% of the whole high dimensional features are crucial to higher classification accuracy, so it is possible how to determine what those best features are or further reducing the feature space dimensions which are the interesting issues to be further continued.

Keywords: *Uyghur, Text classification, KNN, Stop words*

1. Introduction

The text is the most basic information carriers, classification is a supervised machine learning methods, and text classification is the key technologies to obtain and organize large amounts of text data. Research and implementation of text classification is a basic task of natural language understanding and machine learning. With the rapid development of information construction in Xinjiang west part of china, a lot of text information in Uyghur language and other minority languages presented in digital form, and the text information is growing continually, or a vast amounts of paper-based text information which is accumulated in the past begin to stored in digital form, many application areas require to use a computer automatic classification method to integrate and effective use mass of text messages. So, how to automatically classify a large number of text data of minority languages has become an important research topic of natural language processing of minority languages in Xinjiang, including the Uyghur language.

Automatic text classification mainly divided into two kinds of classification and clustering methods, in which classification (text categorization, noted as TC) is a supervised learning process, which based on a tagged training documents find the relational model between the document features and document class, and then use relational model from this study judging the new document's category [1]. In Chinese and English, the technical methods have already been matured, but text information in Uyghur language and other minority languages are differ from Chinese and English, we cannot directly apply the existing Chinese and English methods, which requires carried out systematic theoretical and specific algorithms, simulation experiments and evaluation according to the different language features. On the basis of this, it is necessary and possible to make some improvements or optimizations. Due to Uyghur text classification research started relatively late, the recall rate of those researchers such as Prof. Wenira team[2] and Dr. Alimujiang [3] and others in the Uyghur text classification are all about 70% (the experimental results based on a small amount of data).

In this paper, based on large-scale text corpus, with c # development platform, we designed and implemented the Uyghur text classification system based on the KNN algorithm as shown in Figure 1, and the test and evaluation results are given below.

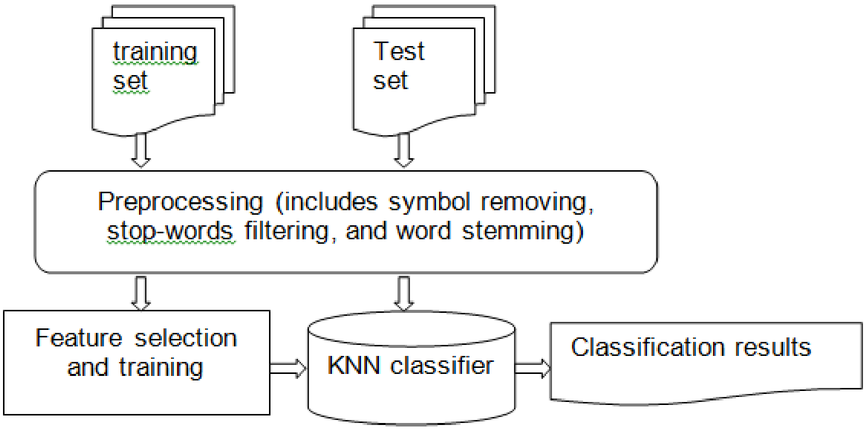

training set
Test set
Preprocessing (includes symbol removing, stop-words filtering, and word stemming)
Feature selection and training
KNN classifier
Classification results

Figure 1. Uyghur Text Classification System based on the KNN Algorithm

2. Uyghur Text Preprocessing

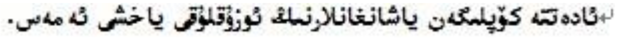
2.1. Uyghur Text Features

Uyghur language belongs to Turkic family of Altaic language system. Its features are as follows: (1) the writing direction of Uyghur is from right to left, from top to bottom. (2) Uyghur has all of 32 letters in which some characters borrowed from Arabic and Persians. (3) Uyghur text completely different from the Chinese and English, is an agglunatative language. In this type of language, the word (word) is the smallest independent language unit. (4)Uyghur word is constituted by a root or stem in addition of other word formation affixes. Word stem is the rest of ingredients after the affixes are removed, and it contains lexical meaning of the whole word.

2.2. Uyghur Text Preprocessing

After the text corpus are collected and are stored in appropriate format the preprocessing procedures are needed. This is the first and most important part of Uyghur text classification, which includes text denoising (identification and removal of non-Uyghur characters, stop word filtering), stemming and so on.

In classification, words are treated as the basic feature unit. Therefore, the text segmentation and word selection from the content of text corpus are the key issues of text representation, which is also the difficult problem of Chinese text processing. But in case of agglunatative language like Uyghur, the word segmentation is not a technically difficult problem, in which word segmented by natural delimiters (spaces) between words, and it is easy to implement. Such as:


ئادەتتە كۆپلىگەن ياشانغانلارنىڭ ئۇزۇقلۇقى ياخشى ئەمەس.

These text is composed of six words are separated by five spaces. For Uyghur, the word segmentation is not a key just mentioned above, but its difficulty is remains in stem segmentation, the procedure of extraction of the true meaning of vocabularies which are taken as feature items. If we take the stems as the feature items, then we can reduce the dimensions of features sets very effectively. If a text formed by the words of same stem and different configuration suffixes such as

ئۆيدە، ئۆيدىن، ئۆينىڭ، ئۆيگە، ئۆيى

(at home, away from home, home, go home, around the house), The essential meaning of these words lies in its stem“ ئۆي”（home）. If we take a word as a feature, then the dimension of the text above is 5. If we take a word stem as a feature, then the feature dimension is dropped to 1.

And on the other hand, if we take the stems as the feature items, then we can effectively eliminate the negative influences of configuration affixes in similarity computing. If the 5 words above are seen from the view of whole word, then they are considered as completely different feature items. If from the view of the word stems, then they are all just considered one feature item, and there are have certain correlations between the texts where those word stems occurred.

Some words are of high frequency of occurrences in the text corpus, but they have little or no influences on the information carried by text. For example, in the English “a, the, of”, in the Chinese “的,了,着”, and strings such as “http”, “.com” and various punctuation, etc. Such words are called stop words. In order to filtering stop words, we need to prepare a stop word list in advance. The stop words list is composed of words those are not too big effect to text representation and those words of appears with equally large of frequencies. In this paper, we combine manual work and statistical methods have established the stop words list containing of 280 stop words. Some of them are shown in Table 1:

Table 1. Stop Words List

words	Common stop words
Auxiliary word	ئال، ئۆت , يۈر, ئەت, كەل, بەر , ئەمەس, ئىمىش, ئىكەن, ئىدى, ...
conjunctions	ھەم, يەنە, بىلەن, ياكى, ۋە, ...
adverb	جىق, ھېلى, ئاران, ئىلگىرى, كېيىن, بەك, چاپسان, ئىلدام, ...
Measure word	تۈپ, قېتىم, كىلوگىرام, ...
pronoun	ئۇلار, ئۇ, سەن, مەن, ...
numeral	مىڭ, ئەللىك, بىرىنچى, بەش, ئۈچ, ...
interjection	ۋاي, پاھ, ئاھ, ۋاھ, ...

After stop words filtering, we can further achieve the goal of accurate representation of text content. In order to achieve the better classification results, after the filtering of stop words, stemming and removing of non Uyghur words procedures, the feature extraction procedure is implemented continually.

2.3. Feature Selection

Feature selection is the procedure of selecting some relevant features those most representative statistical characteristics from a number of original features according to a criterion. The purpose of feature selection is reducing to the dimensions of feature sets, removing redundant features, retaining more category distinguishing features. And the selection criteria generally are removing the common features and limited features to category distinction at large.

Currently, there are a variety of feature selection algorithms are used for research in automatic text classification, but these algorithms has its advantages and disadvantages, there is no accepted optimal method as well. For a specific system, there is a need to compare the effectiveness of those algorithms to determine the optimal method. Common feature selection methods are as follows: document frequency (DF), information gain (IG), mutual information (MI), x^2 statistic (CHI) [4].

In this paper, we have used the x^2 statistical method for feature selection. The x^2 statistical method measure the degree of correlation between word t and document category k, and assuming that the relations between the t and k obey one-degree-of-freedom of x^2 distribution. If word item t has the higher x^2 statistic value for certain category, and the greater the correlation between this word and that category, and the more the information of that word carries about that category. Thus, the x^2 statistic value of a word item t to category k calculated by the formula as follows [5]:

$$x^2(t,k)=\frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \tag{1}$$

Where A is the number of documents containing the t and belong to c; B is the number of documents include t, but do not belong to the k; C is the number of documents not include t, but belongs to the k; D is the number of documents not included t and does not belong to the k; N is the total number of documents. For multi-classification problem, for a word t calculate the x^2 statistic value to every category respectively, and use following formula make the calculation of x^2 statistic value for a word entry t within entire text corpus,

$$x^2\max(t)=\max_{1<i<m}\left\{x^2(t,k_i)\right\} \tag{2}$$

m stands for number of categories.

In order to prove stemming actually have the effectiveness on feature dimension reduction, the contrastive results of original feature dimensions and reduced dimensions on different feature sets of 5, 10, 15 and 20 class documents are given in the Figure 2.

The results show that the stemming procedure effectively reduces the feature dimensions indeed, and stemming is necessary procedure to reduce the feature dimensionality for large-scale text classification.

3. Text Categorization Algorithm

Classification algorithm is the core content of text classification. In addition to this, the text preprocessing, text representation and feature selection constitute the important parts of text classification. Classification algorithm plays decisive role for text classifier’s performance. Currently, the main text classification algorithms came from the two areas of machine learning and statistics. The common algorithms of classification are centroid-based text classification algorithm, naive Bayes classification algorithm, k nearest neighbor classification algorithm, decision tree classification algorithm, neural network classification algorithm and statistical approach based SVM classification algorithm, *etc.*, we have carried out experiment of text classification based on Naive Bayes algorithm, and have obtained the preliminary results [6]. This paper selects K neighborhood (KNN) classification algorithm for text classification task, and have obtained the comparative results of above two different text classification algorithms.

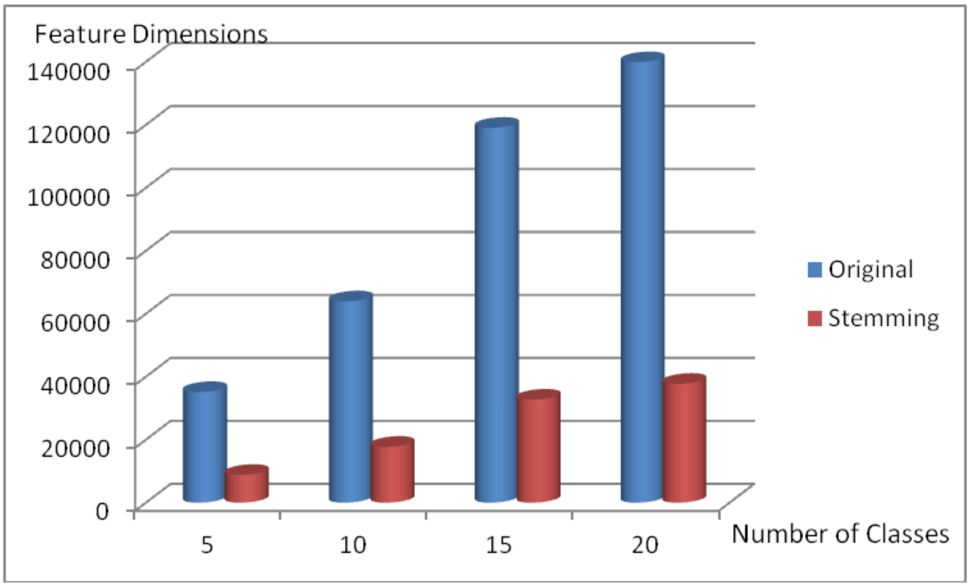


Figure 2. Stemming Effectiveness on Feature Dimension Reduction

KNN classification algorithm is one of the most basic algorithms based on the instance of learning. And it is also one of important methods of pattern recognition of non-parameters. The basic idea is: for a test text, calculate its similarity with each text in the training set; According to the text similarity, to find k most similar training text; then allocate the scores for each text classes, and the score value is the sum of the similarity value between the test text and k training text that belong to given classes; and sort by score value, determine the category of test text.

For a text classification, one can assume that all handwriting samples corresponding to a point in the n dimensional vector space. For a test text, it is need to calculate its similarity between each text in the training sample set. In the nearest neighbor method, we need to calculate the distance between the test sample point and representative all points, and determine the category of test text sample according to the category of most near points. In order to overcome the high mistake rate defects of this method, the nearest neighbor extended to the k neighbor in which the k representative points are selected according to the distance, and determine the category of test text by what kind of category the most points are belong to. In other words, for a given test text to be classified, we can allocate a score as one of candidate categories according to the category of those neighbor points belong to. One can take the similarity between the training text and test text as the classification weights of the categories of that training text belongs to. In these k neighbors, if most texts are belong to same category, then the summation of weights of those neighbors can be considered as similarity between the test text and that category. By sorting the candidate category score values, for a given threshold, one can determine the category of test text. The KNN decision rules can be written as [7]:

$$y(x, C_j) = \Sigma_{d_i \in KNN} \, sim(x, d_i) \, y(d_i, C_j) - b_j \tag{3}$$

Among them, the $y(d_i, C_j)$ indicates that whether the given text d_i belongs to the category C_j (Yes, y=1; No, y=0), And $sim(x, d_i)$ represents the similarity between test text x and training text d_i, where d_i is one of k-nearest neighbors of x, and b_j is the decision threshold. Generally, similarity function $sim(x, d_i)$ calculated by the cosine value of vector angle,

$$\mathrm{sim}(x, d_i) = \frac{\sum_{k=1}^{m} w_k \times w_{ik}}{\sqrt{(\sum_{k=1}^{m} w_k^2)(\sum_{k=1}^{m} w_{ik}^2)}} \qquad (4)$$

Where, m is the dimensions of a feature vector, w_k stands for k^{th} dimension of a feature vector.

Advantages of KNN method are intuitive and easy to understand and apply to, and it is very effective in practical applications, and it is currently one of the most widely used text classification algorithm. But, the disadvantage of KNN method is also very obvious, that is, the high computational cost, due to the distance calculations between test text and each training text, and the single test text time complexity is about m*n where m is the dimensions of feature space and n is the number of training samples. In addition to these, when using KNN method, it is required reasonably choose k, the selection of k value largely determines the classification performance is good or bad. In actual applications, generally, we can choose relatively optimal value of k by adopting the method of cross validation.

4. Text Categorization Experiments and Analysis

4.1. Data Sets

For the Chinese, English text classification and text clustering studies, there have relatively large, standard and open a text corpus available in and out China, so you can carry out the comparative study on performance based on different feature selection and classification methods for same text corpus. But the text classification and clustering research has just started in Uyghur, and there is no standard, large open text corpus available. So, first of all, we have collected the 3000 articles from Uyghur sites on the internet, its content belongs to traffic, characters, sports, health, military, real estate, tourist attractions, economy and computer *etc.*, the category number of that documents is 20, each category have 300 text documents. Then, we have chosen 10 categories contains total of 3000 documents, and randomly reselect the 220 text documents for each category in a total of 2200 documents are taken as the training sample, the remaining 80 text documents in a total of 800 documents are taken as the test sample.

4.2. Evaluation Parameters

The commonly used evaluation parameters including accuracy, precision and recall rate (recall) and F1 value, *etc*.

P (accuracy) = the number of text documents classified correctly / the number of text documents classified

R (recall rate) = the number of text documents classified correctly / the number of text documents due

F1=(2*P*R) / (P+R)

4.3. Experimental Results Analysis

In this paper, using c # programming language, we designed and implemented the Uyghur KNN text classification system. The hardware configuration includes the Intel dual-core E7300 2.66 GH processor, 2GB of memory for PCs; the operating system is Windows 7. When text classification conducted using KNN classifier, we will get different results for different k value, so choose the appropriate k value is very important. Figure 2 is the classification results on the different k values. The results showed that we have got better classification results on the k value about 20. Therefore, all in the following experiment, we take the values of k is 20.

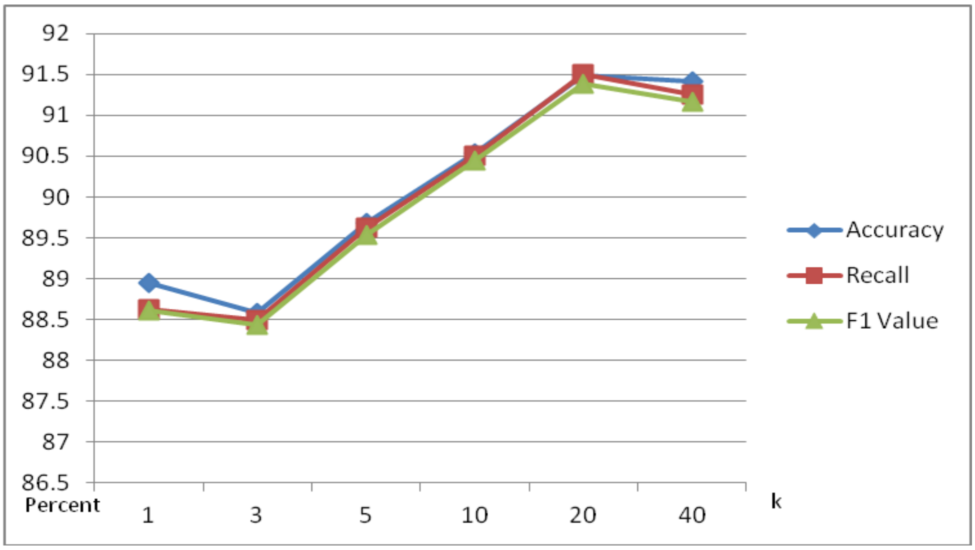


Figure 3. KNN Classification Effects under the Different K Value

In this experiment, the test text corpus is belongs to the following 10 categories, the classification results and efficiency of KNN classification algorithm are shown in Table 2,

Table 2. Detailed Experimental Results of KNN Classification

category	Classification result			Classification effectiveness		
	Original text number	Classified correctly	Actual classification	P (%)	R (%)	F1 (%)
traffic	80	68	97	70.10	85.00	76.84
sports	80	80	88	90.91	100.00	95.24
health	80	67	77	87.01	83.75	85.35
house	80	76	85	89.41	95.00	92.12
military	80	31	45	68.89	38.75	49.60
education	80	74	92	80.43	92.50	86.05
tourist	80	72	91	79.12	90.00	84.21
custom	80	49	70	70.00	61.25	65.33
economy	80	70	97	72.16	87.50	79.10
computer	80	65	84	77.38	81.25	79.27
average				**78.54**	**81.43%**	**79.31%**

The average efficiency of KNN classification lower than Naive Bayes classification algorithm, but the classification accuracy and classification speed of KNN and NB classification algorithm different from each other, details are given in Figures 4 and 5,

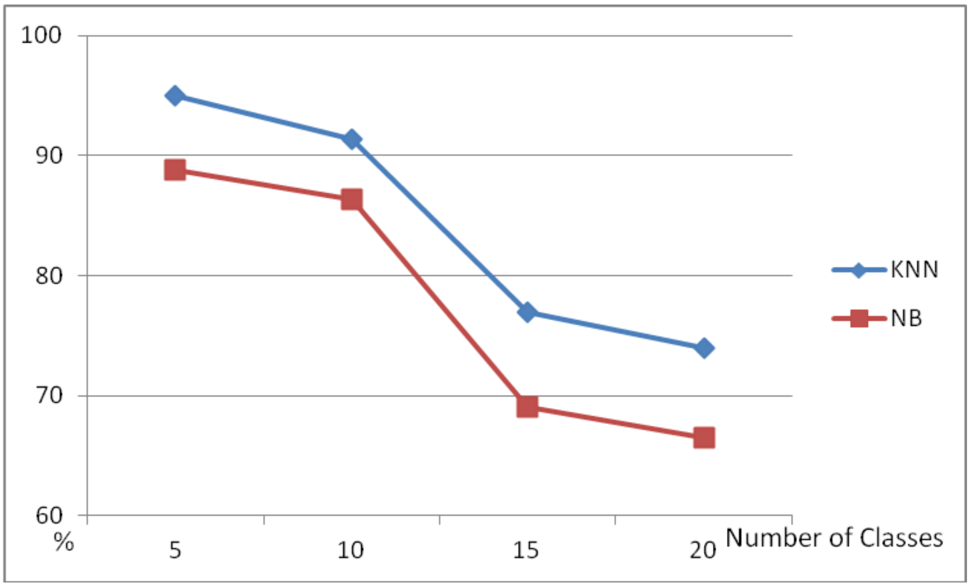


Figure 4. Comparative Classification Results of KNN and NB Classifier

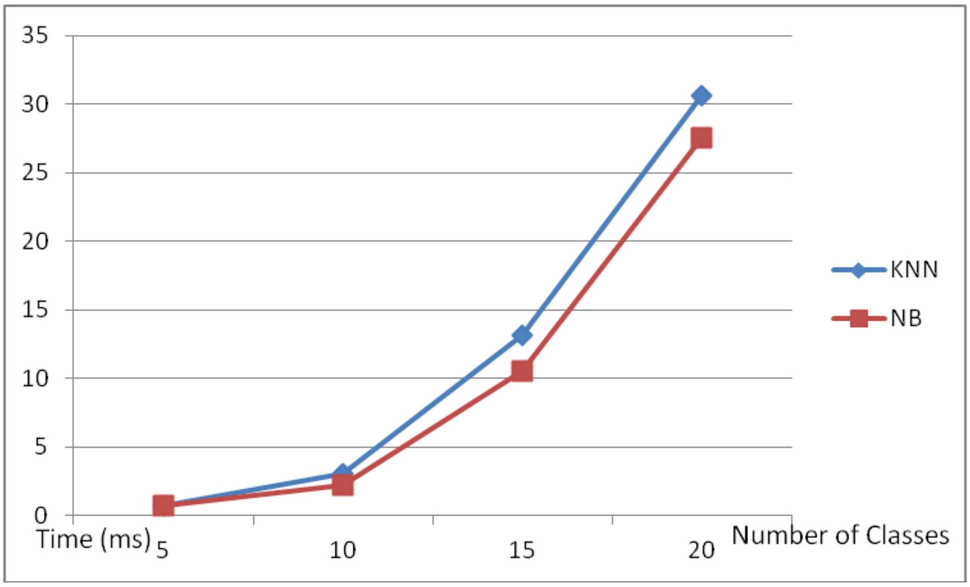


Figure 5. The Time Efficiency of the KNN Compared to NB Classifier

In addition, select the different number of features by using x^2 statistical methods and the classification results on corresponding different feature sets are also given in Figure 6.

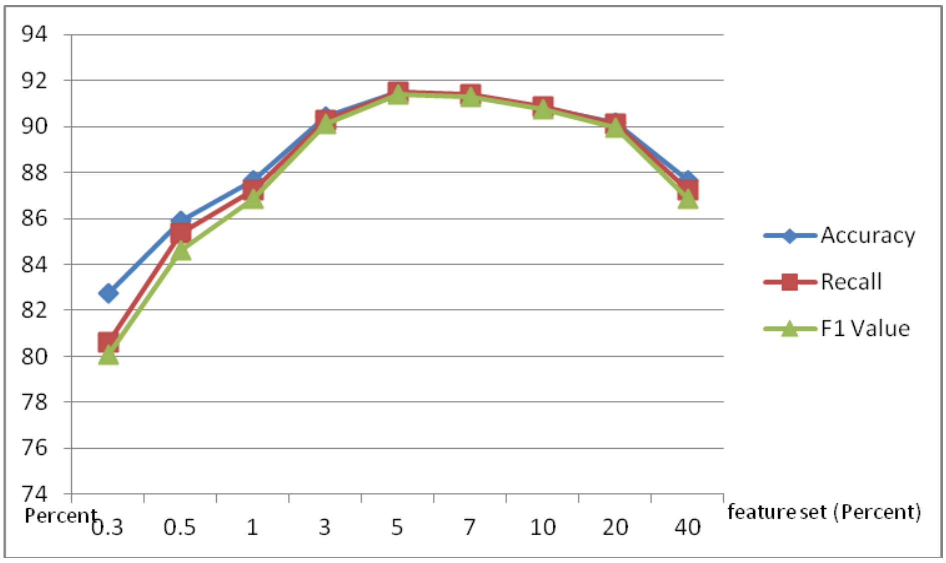


Figure 6. Classification Efficiency under Different Percent of Features

It is also can be seen from the Figure 6, the experimental results of KNN classification algorithm based on different percentage of feature sets are different, and for some of the feature sets the classification results are good, and for some other feature sets the classification result are bad. Very few features are not really representative for the text content; the too many feature dimensions will lead to feature dispersion, and the 3%-5% feature sets are ideal to Uyghur text classification.

5. Conclusions

This article described the characteristics Uyghur, text preprocessing and text representation method, studied the KNN classification algorithm more in-depth, and got some conclusions by a combination of theoretical analysis and experimental methods. On the basis of large-scale text corpus, with the method of the KNN, the Uyghur text classification experiments are carried out. It is proved that after the preprocessing, the only 3%-5% features sets are crucial to text classification. And How to find those features and further reduce the feature dimensions are the further research contents of following paper.

Acknowledgements

This work has been supported by the National Natural Science Foundation of China under grant of 61163033.

References

[1] K. P. Soman, “Data mining based tutorial”, translated into Chinese by Fanming and Niu chang yong, China Machine Press, Beijing, (**2013**).

[2] W. Zhen and W. Musa, “Automatic classification technology of search engine results”, Master’s Thesis, Xinjiang University, Urumqi (**2010**).

[3] A. Aysa, T. Ibrahim, H. Omar and M. Ali, “The Uyghur text classification based on machine learning research”, Computer engineering and application, vol. 5, (**2012**).

[4] L. X. Fei and Lijun, “Data mining and knowledge discovery”, Higher Education Press, Beijing, (**2003**).

[5] Z. Pengzhao, “Chinese text classification feature selection method based on X^2 statistics research”, Doctoral Dissertation, Chongqing University, Chongqing, (**2008**).

[6] A. Ablat, T. Tuhti and A. Hamdulla, “Naive Bayes based Uyghur text classification algorithms and performance analysis”, Computer Applications and Software, vol. 29, no. 12, (**2012**).

[7] W. Xiaoqing, “Chinese text classification feature selection method research”, Southwest University, Xi’an, (**2010**).

Authors

Palidan Tuerxun received her M. S. degree in 1996 from Liaoning University, China. She is currently working toward PhD. degree in Northwestern University, China. Since 1992, she has been working as a teacher at Xinjiang University, and since 2004, she was an associate professor in school of software of Xinjiang University. Her research interests are machine learning and Uyghur natural language processing.

Fang Dingyi currently is a Professor in the School of information and technology, Northwestern University, Xi’an, China. His research interest includes networking and information security.

Askar Hamdulla received B.E. in 1996, M.E. in 1999, and Ph.D. in 2003, all in Information Science and Engineering, from University of Electronic Science and Technology of China. In 2010, he was a visiting scholar at Center for Signal and Image Processing, Georgia Institute of Technology, GA, USA, tutored by Professor Biing-Hwang (Fred) Juang. Currently, he is a professor in the School of Software Engineering, Xinjiang University. He has published more than 120 technical papers on speech synthesis, natural language processing and image processing. He is a senior member of CCF and an affiliate member of IEEE.