Research of Feature Selection for Text Clustering Based on Cloud Model

Junmin Zhao
Henan University of Urban Construction/Institute of Computer Science and Engineering, Pingdingshan, China
zhaojunminhncj@yeah.net

Kai Zhang
Henan University of Urban Construction/Institute of Computer Science and Engineering, Pingdingshan, China

Jian Wan
ZhengZhou ShiYi Technology Co. Ltd, Zhengzhou, China

***Abstract*—Text clustering belongs to the unsupervised machine learning, the discriminability of class attributes cannot be measured in clustering. And the traditional text feature selection methods cannot effectively solve the high-dimensional problem. To overcome the weakness in existing feature selection, this paper proposes a new method which introduces the cloud model theory into feature selection, constructs the clouds filter for clustering documents. The distribution of document words is constructed in a microcosmic level. By employing the cloud model digital characteristics we can better compute the separability between feature words. Experimental results with K-means algorithm show that our method can remarkably improve the accuracy of text clustering.**

***Index Terms*—feature selection, cloud model, TF-IDF, K-means algorithm**

I. INTRODUCTION

Clustering is an important data mining technique which groups the data objects into multiple classes according to the similarity information; the data object set has high similarity within the same cluster and the differences between data objects in different clusters as large as possible. Text clustering technology produces along with the large-scale text data; it is commonly used in information retrieval field. It is different from the traditional clustering in this point that clustered objects are unstructured text data. In order to form mathematical model that the computer can deal with, the document needs to be processed as 1) word segmentation, 2) removing the stop words, 3) feature selection and so on, finally some appropriate clustering algorithms can be used on the reduced-dimensional feature space.

Text data is usually described with the vector space model. After preprocessing, each document is transformed into a feature vector; each word is viewed as one dimension in the feature space. Since a document contains several different words, which makes the text vector space dimension very high and sparse. The high-dimensional and sparse features bring great noise to the text clustering and make clustering performance fall dramatically, so there is an urgent need for dimension reduction. So far the most direct dimension reduction way is feature selection. Feature selection means selecting a small part of the most effective features from the original set[1,2], it focuses on studying the word importance of evaluation function. The process of feature selection consists of two steps, first, calculating the importance value for each word, and then select the words whose with the more importance value than a certain threshold. There are many traditional feature selection methods such as Document Frequency, Information Gain, Mutual Information, Chi Square Statistics etc. These traditional methods cannot be effectively used in clustering [3]. Many scholars deeply studied the feature selection, George Kingsley Zipf put forward the famous Zipf's law [4] on the basis of making large-scale statistics on word in the English literature and studying the law in them. In 2009, Estevez, P.A.[5] proposed the normalized mutual information feature selection method (NMIFS) which is proved superior to the traditional. Purdue university's Jennifer G.Dy [6] proposed an unsupervised learning feature selection method which used packaging framework for feature selection. Salton[7] used *TF-IDF* method to evaluate the importance of a feature word in a document of the corpus set. Yu Fang [8] combined the *TF-IDF* and mutual information methods for feature selection and got a better result. Lou Haifa etc. Ref.[9] used the *CHI* and *TF* two methods at the same time in the feature selection, this method makes up for the deficiency of the *CHI* which ignores the word frequency of feature word itself.

This paper makes a deeply study on feature selection in text clustering, our method employed the uncertainty of cloud model theory [10], and combined fuzziness and randomness in the concept of uncertainty. Our method is designed in three steps: First, the cloud model theory is used to map the feature words into clustering word cloud droplets. And then, the word cloud droplets will be

Manuscript received January 9, 2013; revised March 12, 2013; accepted May 5, 2013.

doi:10.4304/jsw.8.12.3246-3252

transformed into clustering document cloud. Finally, we will build document clustering cloud filter, which can select the feature words with high separability. Experiments results show that the proposed method is effective, the clustering result is more accurate than the traditional feature selection methods with K-means clustering.

II. Analysis of Traditional Feature Selection Methods

To improve the accuracy of text classification, classical feature selection methods such as mutual information (*MI*), χ^2 Statistic, information gain (*IG*) and *TF-IDF* are studied, these methods are briefly described as follows.

A. Mutual Information

Mutual information is the result of information theory [11]; it is used for measuring correlation of two objects [12]. In text feature selection, mutual information refers to correlation degree between a certain word and a class. When mutual information between a word and a class is greater than a certain threshold, then we think the word is well associated with the class; otherwise, the word is not associated with the class. Mutual information is defined as following:

$$MI(t_i) = \sum_j p(C_j) \log \frac{p(t_i / C_j)}{p(t_i)} \tag{1}$$

In formula (1), $p(C_j)$ represents the prior probability of class C_j in the whole training set, $p(t_i / C_j)$ represents proportion that documents containing item t_i in class C_j, while $p(t_i)$ represents proportion that documents containing item t_i in the whole training set.

B. χ^2 Statistics

χ^2 Statistic, also known as *CHI*, assumes the relation between feature word t_i and the certain class is in accordance with χ^2 distribution [13]. Like *MI*, *CHI* needs to quantify the correlation degree between target feature word t_i and C_j class so as to measure the account of class information carried by feature words. The higher the *CHI* value is, the greater the feature word t_i and C_j class is related. In feature selection, we should choose the features of *CHI* with the maximum statistic.

$$\chi^2(t_i, C_j) = \frac{N(AD - BC)^2}{(A+C)(A+B)(B+D)(C+D)} \tag{2}$$

In the formula (2), *N* refers to the documents number in the training corpus, *A* represents the number of documents that include t_i and belong to C_j class, *B* represents the number of documents that include t_i but do not belong to C_j class, *C* represents the number of documents that do not include t_i but belong to C_j class, while *D* represents the number of documents that neither include t_i nor belong to C_j class.

C. Information Gain

Information gain (*IG*) is an effective feature selection method, and is widely used in text data mining. Information gain does not concern the relation between a certain feature word and a certain class, but treats all classes in training set as a whole. And the importance of a certain word is measured by calculating the information amount that each class takes. Information gain of the feature word *t* refers to the *D*-value between the information amount of the whole training set without regard to feature word *t* and that of the training set with regard to feature word *t*. It is defined as follows:

$$IG(t_i) = -\sum_{j=1}^{M} p(C_j) \times \log p(C_j) - \{p(t_i) \times [\sum_{j=1}^{M} p(C_j / t_i) \times \log p(C_j / t_i)] + p(\overline{t_i}) \times [\sum_{j=1}^{M} p(C_j / \overline{t_i}) \times \log p(C_j / \overline{t_i})]\} \tag{3}$$

In the formula (3), $p(C_j)$ represents the proportion that documents of C_j class take in the data corpus, $p(t_i)$, $p(C_j / t_i)$ represents the conditional probability of a certain document belonging to C_j class when the document does not contain feature word t_i, $p(\overline{t_i})$ represents the percentage that documents with feature word t_i take in the training corpora, $p(C_j / \overline{t_i})$ represents the proportion of a certain document without entry t_i belonging to C_j class, and *M* represents the total number of all the classes in training corpora.

D. TF-IDF

According to the main idea that if one word appears many times in the document collection, and rarely appears in other documents, the word has strong class distinction degree. Salton[7] proposed *TF-IDF* method in 1988 to evaluate the importance of a feature word for a certain document in the data corpus. In *TF-IDF* evaluation standard, $tf(t)$ is the frequency of feature word t appearing in the document collection, $idf(t)$ is the macroscopic description of the distribution of feature word t in the document collection. The calculation formula (4) and (5) are as follows:

$$idf(t) = \log(N / n_i + 0.01) \tag{4}$$

$$tf\_idf(t) = tf(t) \times \log(N / n_i + 0.01) \tag{5}$$

For the matter of clustering, *N* is the total number of cluster- testing documents, n_i is the number of documents with feature word *t* in the entire testing set. In order to avoid the $idf(t)$ is zero, we add a smoothing factor as 0.01. *TF-IDF* method is to find those feature words that appear frequently, but the documents that contain these feature words is few. It is just a borrowing algorithm without abundant theoretical basis, but this method indeed reduces the dimension of the feature space[14].

However, the above classical feature selection methods have some disadvantages. Features with low frequency are the main part of *MI*, but we should select features with much mutual information. Since *MI* is beneficial to the low-frequency features, thus it is likely to cause over-fitting; χ^2 statistic method ignores the frequency of feature words in feature selection; Information gain considerers the situations of occurrence and absence of features, but it only examine the contribution of features to the whole system, not to a certain class; Since *TF-IDF* can directly measure the differences between feature words in a document without the class information, it is quite appropriate for feature selection of clustering documents. Aiming at the shortcomings of the existing feature selection methods, we apply the cloud theory into feature selection in text clustering. On the basis of the fuzziness and randomness of uncertain concepts, the cloud filter of clustering documents is used to construct clustering cloud space.

III. Feature Selection Based on Cloud Model

To cluster the texts, a low-dimensional feature set is needed to represent the content of the documents effectively. No matter for what kind of clustering model, feature selection is an indispensable step. This paper considers that a useful feature word should have the high discriminability, which indicates the ability of distinguishing the class that *t* feature word represents from other classes. In order to select feature words which satisfy this requirement, cloud model theory is used. in this paper, first, we construct clustering cloud for documents to produce cloud feature; then, filter each feature word in clustering documents; finally, we will get cloud feature space. The process is shown in Fig.1.

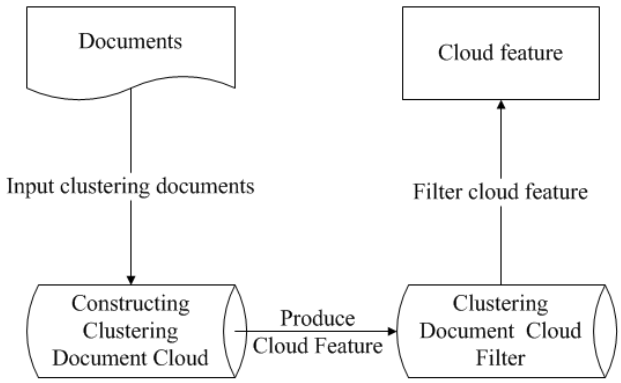

Figure 1. The process of constructing cloud feature space

A Cloud Model

Cloud model [7] as a transformation model is proposed by Chinese prof. Li; it can make conversion between qualitative concepts and quantitative values. Digital characteristics of cloud model can reflect the overall properties of the concept. In order to transform the qualitative from to quantitative one, we can use cloud model to describe some phenomena existing in the natural language, such as randomness, fuzziness and relationship between them. The Cloud is defined as follows: assuming that U is a quantitative domain expressed by exact numerical, C is the qualitative concepts on the domain, x is a quantitative value, $x \in U$ and x is stochastic realization of C, $\mu(x)$ represents the certainty degree of x to C, $\mu(x) \in [0,1]$ is a random variable with stable tendency. If $\mu : U \to [0,1] \quad \forall x \in U \quad x \to \mu(x)$, then, we call the distribution of x on domain cloud, x is called as a cloud droplet. In the cloud model, the numerical characteristics of cloud model are denoted as Expectation *Ex*, Entropy *En* and Super-entropy *He*, and they reflect the whole characteristics of the quality conception *C*. *Ex* is expectation which comes from the distribution of cloud droplets samples on domain space; *En* is a criterion of measuring the uncertainty of qualitative concept, it can reflect the randomness and fuzziness of qualitative concept; *He* is the second-order entropy of the entropy and co-determined by both the randomness and fuzziness of entropy.

The formula (6), (7), (8) are as follows:

$$E_x = \overline{x} \tag{6}$$

$$E_n = \sqrt{\frac{\pi}{2}} \times \frac{1}{n} \sum_{i=1}^{n} \left| x_i - E_x \right| \tag{7}$$

$$H_e = \sqrt{S^2 - E_n} \tag{8}$$

Cloud model uses forward cloud generator and backward cloud generator to transform the qualitative concept and quantitative data, the forward cloud generator transforms qualitative to quantitative, Fig.2 gives the cloud example, the value of *Ex, En, He* is 0, 4 and 0.4 respectively. Conversely, backward cloud generator is used. Due to this paper using the cloud model for feature selection, backward cloud generator [15] is needed. Fig.3 is the transition diagram.

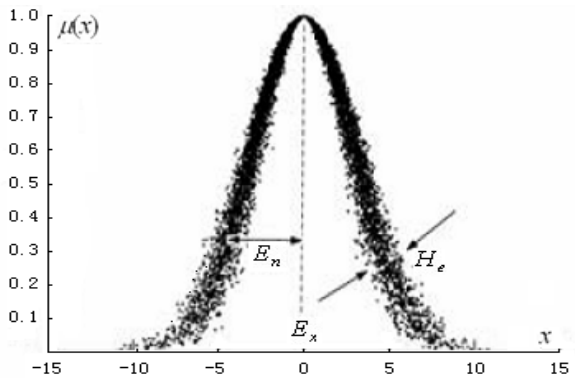

Figure 2. Ex=0, En=4, He=0.4 forward cloud

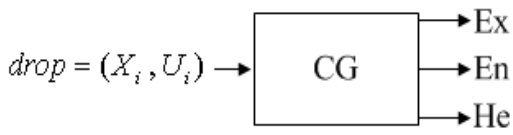

Figure 3. backward cloud generator

The implementation algorithm of backward cloud generation is shown as follows.

Input: sample x_i , and i=1, 2, 3..., n.

Output: three digital features: *Ex, En, He*.

The calculation steps are as follows:

(1) Calculate sample mean based on x_i Use(9);

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{9}$$

(2) Calculate first-order sample absolute center matrix based on x_i , $\overline{X}$ Use(10);

$$\frac{1}{n}\sum_{i=1}^{n} | x_i - \overline{X} | \tag{10}$$

(3) Calculate sample variance Use(11):

$$S^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \overline{X})^2 \tag{11}$$

(4) Using the result in step (1), (2), (3) and formula (6), (7), (8) calculate E_x, E_n, H_e .

B Constructing Clustering Document Cloud

The aim of constructing clustering document cloud is to choose the words with high distinguishing ability in documents without class labeling. In the process of clustering, each clustering document $d[i]$ needs to be mapped to a single point $p[i]$ in the unified feature space V . $V : d[i] \to p[i]$, $d_i \in DOC$ DOC represents the whole test sets. To some extent, the process of building space V is the process of completing feature selection for clustering document. A reasonable V can make the points p through the distance between them reflect their similarity between the documents d they representing. In fact, the clustering document is distinguishing in content, If extracting representative features from test document to build the unified space V is possible, then the document with similar meanings can be clustered more accurately.

Here we use cloud model to carry out the feature selection, firstly document clustering cloud need to be built, the steps is as follows:

Step1: Preprocess each document in test set with word segmentation technology, then removing the stop-words in documents;

Step2: Calculate each different word in the test set, and then establishing the original initial feature set V_SET ;

Step3: For each feature word t in V_SET , traverse each document $d[i]$ in *DOC* and calculate the frequency of *t* in each document *d*.

Here feature word frequency consists of droplets $drop[i]$, $i \in N(DOC)$; meanwhile, the droplets all together form clustering document clouds.

In the experiment, we choose "art", "agriculture", "model", "computer", "sports", "fund" as feature words to evaluate the their distinguishing ability in different documents. As shown in Fig. 4, the experiment result shows the clustering document cloud can capture feature words with high separability. In can be noted that the horizontal axis represents each document in 300 corpus, vertical axis represents the word frequency of the six feature words appear in the 300 documents.

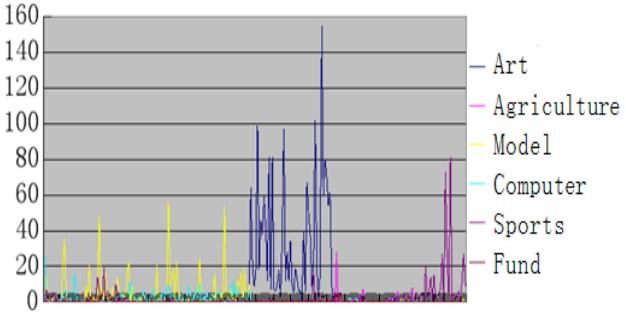

Figure 4. The cloud distribution of "art", "agriculture", "model", "computer", "sports", "fund" in 300 different documents

C Clustering Document Cloud Filter

In text clustering, documents are often represented by Vector Space Model (*VSM*). Suppose there are two documents named *d1* and *d2*, which are represented by *v1* and *v2*. The similarity between *d1* and *d2* can be measured by the cosine value of *v1* and *v2*. The formula (12) is as follows:

$$sim(d_1, d_2) = \cos(v_1, v_2) = \frac{v_1 \bullet v_2}{|v_1 \bullet v_2|} \tag{12}$$

Document vector consists of different words which cause the very high dimensional feature space. In this way, it takes a high cost to calculate similarity. Due to the existence of some useless features, error of similarity calculation can be produced. Assume document *A* and *B* are short documents of the same category. Document *A:* Wang Ming thinks the performance of this computer is excellent. Document *B:* Li Lei also thinks the performance of this computer is excellent. If we need to select effective features of these two short documents, there will be computer, performance and excellent in feature space *V*. Thus, document *A* is represented by (1, 1, 1), and document *B* (1, 1, 1) as well, and the number 1 refers to word occurrence in the document. Then the similarity between document *A* and *B* is 100%, but the similarity is much lower if we don't select features. This example indicates the importance of feature selection in text clustering.

Furthermore, the dimension of original feature space V_SET we get is too high, which seriously affect the clustering effect. In order to form a reasonable feature

space V_SET with lower dimension, we build a clustering document cloud filter based on the cloud model theory called *Cluster_Filter*, which can select appropriate feature words. Our approach is designed with these points: if feature words have low frequencies or keep a relatively stable distribution in each documents, such feature words do not have class attributes, and only those which satisfy the two conditions are the important words we prefer to select.

Therefore, the *Cluster_Filter* must obey following principles:

(1) The feature words in the whole clustering document *DOC* should possess high frequency;

(2) The frequency of feature words in the different clustering document *d* should have obvious fluctuation.

Based on the above two points, we use the digital characteristics *Ex*, *En* and *He* to measure the "importance" of a feature word, the formula (13) as follows:

$$important(t) = d\_E_x(t) \times d\_fluctuation(t) \quad (13)$$

The first factor $d\_E_x(t)$ is expectation for feature words, which reflects the frequency; the second factor $d\_fluctuation(t)$ is a linear combination for *En* and *He*, which reflects the fluctuation of feature words. The formula (14) as follows:

$$d\_fluctuation(t) = \alpha \times d\_E_n(t) + \beta \times d\_H_e(t) \quad (14)$$

Formula (14) indicates that the two features are reflected at the same time. In the experiments, the ratio of *a* and *b* is 3:1, which is an empirical value.

IV. Experiment and Analysis

A Experimental Descriptions

In order to verify the effectiveness of the new method, we compare the proposed method with *TF-IDF* by adopting K-means clustering algorithm. In the unsupervised machine learning, it is difficult to determine the center value (*k*) of clustering. In the experiments, we specify the same clustering initial points in the generated feature subset by the two methods respectively, and use the K-means clustering algorithm to compare with the clustering effect. The experimental dataset including 300 documents is collected from Fu Dan university' Chinese corpora. The dataset covers many fields such as literature and art, computer, history, economy, aviation, sports; we randomly select 50 documents in each class.

B Experimental steps

Preprocessing stage: clustering has no training corpora, so we must carry out word segmentation and remove stop-words from each document, then calculate each different word in the dataset and establish the original initial feature set V_SET;

Feature selection stage: the main task of this stage is dimension reduction. Here, we use *TF-IDF* and *Cluster_Filter* to generate feature words respectively, the size of feature sets is 10, which consists of 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1500 feature words respectively;

Document representation stage: calculating similarity between the documents is the core of text clustering algorithm based on K-means. So, all documents will be transformed into the space vector form. In the space vector, each dimension represents a feature word, and the value of each vector coordinate is the frequency of a feature word in this document. For instance, there are two document fragments: *d1*: 'My favorite sport is basketball, and he likes playing basketball too.' *d2*: 'This basketball is great and feels good, I am going to play.' After preprocessing, the V_SET is (sport, play, basketball, feel). Consequently, the corresponding vector *V* for *d1* is (1,1,2,0), and *d2* (1,1,1,1).

K-means clustering stage: K-means algorithm is the most representative hierarchical clustering algorithm by measuring distance between documents, and it is widely applied in text clustering field. K-means algorithm takes euclidean distance to measure the similarity of documents in vector space. It is considered that the closer two texts are, the higher similarity between them is.

Input: the value of *k* and text[n];

(1) Select *k* initial clustering points, for instance, t[0]=text[0], t[k-1]=text[k-1];

(2) Calculate the similarity between text[k]...text[n] and t[0]...t[k-1]; suppose the similarity between text[k]...text[n] and t[i] is the highest, then mark it as *i*;

(3) For points marked as *i*, we need to calculate t[i] repeatedly, which is the quotient of the sum of all text[j] marked as *i* dividing the number of texts marked as *i*;

(4) Repeat step 2 and step 3 until the criterion function converges, and make the clustering center change little.

The criterion function (15) is defined as follows:

$$f = \sum_{i=1}^{k} \sum_{t \in C_j} \left(\left| p - m_j \right| \right)^2 \quad (15)$$

In the formula above, *P* denotes the document which needs to be clustered, m_j is the center of cluster C_j. Fig. 5 shows the process of calculating similarity in the clustering, red and yellow dots on behalf of the documents to judge, *a* and *b* represent two clusters which have been clustered.

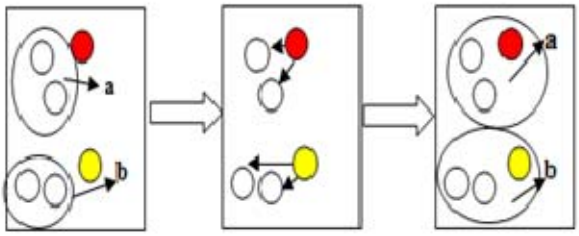

Figure 5. The process of calculating similarity in the clustering

C Result and Analysis

To evaluate the clustering result, we use the accuracy rate and recall rate as the evaluation standard. Accuracy rate and recall rate are two frequently referred concepts and indices used in data mining and search engine in

internet. For information retrieval, the more the contents that are retrieved, the higher the recall ratio is; and the larger the recall rate is, the better. On the other hand, accuracy rate requires more relevant documents that are retrieved and higher value of precision. Accuracy rate refers to the rate of documents that are correctly clustered, while recall rate indicats the rate of the target categories that are recalled from focus fields.

Computational formula of accuracy rate and recall rate (16) (17) are defined as follows:

$$\Pr ecision = \frac{m}{m+n} \tag{16}$$

$$\mathrm{Re}call = \frac{m}{m+o} \tag{17}$$

In the formula(16) (17), *m* represents the number of documents that are clustered to a certain class; *n* represents the number of documents that are clustered to a certain class by mistake; *o* represents the number of documents that are supposed to be clustered to the certain class, but are clustered to other classes by mistake.

Both accuracy rate and recall rate are important criterions for evaluating the performance of models. It can be seen observed from the formula above that there is no necessary relevance between accuracy rate and recall rate, but they interact each other in large-scale data. Therefore, take all things into consideration; we need to find a balance point between them. We adopt the *F-measure* method proposed by Van Rijsbergen in 1979 to comprehensively consider the accuracy rate and recall rate of the clustering results, the calculation formula (18) is as follows:

$$F-measure_{\beta} = \frac{(\beta^2+1)\,precision \times recall}{\beta^2 \times (precision + recall)} \tag{18}$$

By adopting *TF-IDF* and *Cluster_Filter* feature selection methods, we select 11 feature subsets with different dimensionalities, and perform the clustering experiment on these 11 feature sets by K-means clustering method, the result is shown in Fig.6. The vertical axis represents F-measure value, and the horizontal axis represents the size of feature set.

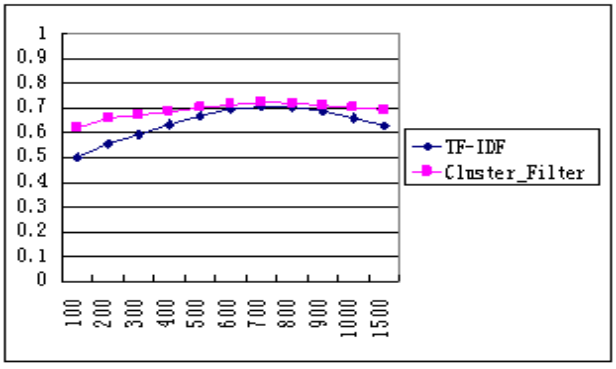

Figure 6. Comparison of F-measure values between *TF-IDF* and *Cluster_Filter*

D Experimental Results

As shown in Fig.6, the performance of *Cluster_Filter* method is superior to *TF-IDF*, when using K-means clustering algorithm on the 11 different feature sets respectively. Especially, for the feature sets are less than 600 words, the *F-measure* score of *Cluster_Filter* is obviously higher than *TF-IDF* . Besides, if we choose the 100-feature words, the former's *F-measure* will reach 10.1 percentage points higher than the latter. The *F-measure* values of the two methods increase as the increasing of the number features, which reaches the peak as the size of feature set reaches 700. The *F-measure* values will decrease when the number of feature set increases by more than 800; the reason is that redundant features have been selected as the feature set increasing.

Fig.7 shows the part of the feature words, which are produced by *Cluster_Filter* and *TF-IDF* method respectively. As far as feature words are concerned, we can observe that feature words produced by *Cluster_Filter* carry more class information than *TF-IDF*.

Cluster_Filter	history, art, sports, novel, economy, literature, research, society, development, culture, enterprise, works, system, value, problem, thought, art-history, aviation, agriculture, technology, Lu Xun, data, expression, process, method, creation, service, construction, spirit, theory, measurement, training, figure, world, market, form, management, tradition, subject, woman, information, image, activity, computer, relation, life, influence, revolution…
TF_IDF	musical, communication, layer, transformation, Deng Xian, connection, opera, mining, strain, merchant, fault, analysis, aspect, object, way, education, function, project, commodity, modern, athlete, literature and art, movement, need, process, supply, realization, basis, nature, importance, existence, reality, doctrine, creation, times, among, original, think, content, program, time, effect, town, consciousness, art, historiography, human, construction, Wang Zhaojun, stock...

Figure 7. The part of the feature words produced by *Cluster_Filter* and *TF - IDF* (50 words)

In the *TF-IDF* method, *IDF* utilizes the distribution information of the words, but it is a macroscopic distribution not microcosmic one. Therefore, *TF-IDF* method may be not suitable to measure the distinguishing ability of feature words, *IDF* only care about whether a word appears in a document, but it does not consider that how many times the word appears, this leads to the condition that some words with strong discriminability are filtered out since it appears in all documents. Nevertheless, *Cluster_Filter* not only considers the word frequency, but also makes full use of distribution ratio of feature words from the microscopic angle. In fact, even if a word appears in many documents, its frequencies in these documents also have great difference, and the word still has strong separability in the whole text data.

Here we illustrate it with an example:

Assume there are 10 documents needed to be clustered,

word T_m appears in each document on average, while T_n doesn't. Table I shows the frequency distribution of T_m and T_n in 10 different documents with same class.

TABLE I.
THE FREQUENCY DISTRIBUTION OF T_m AND T_n IN 10 SAME CLASS DOCUMENTS

doc[i] i= 1, 2, 3……10										
Tm	40	2	1	1	1	1	1	1	1	1
Tn	5	5	5	5	5	5	5	5	5	5

First, we calculate the values and of *TF-IDF* and importance of T_m. Obviously, the results show that the *TF-IDF* value is equal, while the importance of T_m is higher than that of T_n. Through comparing the value of *TF-IDF* with importance, it is easy to find the deficiencies of *TF-IDF*, while our method can effectively overcome the deficiencies. For example, the word "economy" has a strong distinguishing ability, but many non-economic documents also contain it. Though "economy" is a high discriminative word, it will be likely to be filtered out if using *TF-IDF* for feature selection.

V. CONCLUSIONS

In the unsupervised machine learning process, the traditional feature selection methods cannot effectively solve the problem of feature dimensionality reduction in text clustering. By introducing the cloud model theory, this paper builds cloud filter which uses digital characteristics to represent the discriminative feature words. Since our method focuses on the word distribution information over documents from a microscopic view. Therefore, the importance of feature words can be described more accurately. Compared with the traditional *TF-IDF* method, feature words which are selected by *Cluster_Filter* method contain more class information. Finally, we apply K-means algorithm on the generated feature sets by two methods respectively, results show our method is superior to *TF-IDF* in performance.

REFERENCES

[1] Y .Yang, J.O.Pedersen. A comparative study on feature selection in text categorization. In Proc. 14th Int'l Conf. Machine Learning,1997:412-420.
[2] Liu.T., G.&Chen.Z. An Effective Unsupervised feature selection method for text clustering. Journal of Computer Research & development. 2005,42(3), pp.381-386.
[3] Wang WL, Liu PY ,Liu KF A feature selection algorithm for web documents clustering. Computer Applications and Software.2007,24(1):pp.154-156.
[4] http://en.wikipedia.org/wiki/Zipfs_law, April.2013.
[5] P. Estevez, M. Tesmer, C. Perez and J. Zurada. Normalized mutual information feature selection. IEEE Trans. Neural Netw. vol. 20, 2009, pp.189.
[6] J.G Dy and C.E. Brodley, Feature Subset Selection and Order Identification for Unsupervised Learning, roc. 17th Int"l Conf. Machine Learning, 2000, pp. 247-254.
[7] Salton, G and Buckley, C. Term-weighting approaches in automatic text retrieval. Information Processing&Management 24(5) 1988, pp.513-523.
[8] Yu F, Jiang YF. A Feature Selection Method for NB-based Classifier, Acta scientiarum naturalium universitatis sunyatseni. 2004, 43 (5), pp. 118-120.
[9] Luo HF, Wu G, Yang JS. Way of text classification based on Bayes. Computer Engineering and Design, 2006,27(24), pp. 4746-4748.
[10] Li DY, Liu CY, Du Y, Han X.Artificial Intelligence with Uncertainty Journal of Software 2004 Vol15.No.11, pp.1583-1592.
[11] Shannon CE. A mathematical Theory of Communication. The Bell System Technical Journal.vol.27,July, and Oct .1948. pp. 379-423, 623-656.
[12] Hamming R W. Coding and information theory. Englewood Cliffs, NJ:Pretice-Hall, 1986.
[13] Ted Dunning, Accurate methods for the statistics of surpris e and coincidence,Compu tational Linguistics, v.19 n.1, M arch 1993, pp.61-74.
[14] Muath Alzghool , Diana Inkpen, Clustering the topics using TF-IDF for model fusion, Proceeding of the 2nd PhD workshop on Information and knowledge management, October, 2008, Napa Valley, California, USA.
[15] LU HuiJun,WANG Ye,LI DeYi,LIU ChangYu. The Application of Backward Cloud in Qualitative Evaluation, CHINESE JOU RNAL OF COMPUTERS 26(8), 2003, pp.1009-1014.

Junmin Zhao, born in Henan Pingdingshan in the middle of China, September 1978. He is a teacher of Henan university of urban construction. In 2008 graduated from Henan university of economics and law, major in computer science, received a master's degree in engineering; now study in central China normal university to get computer PhD.

Kai Zhang, born in Henan Pingdingshan in the middle of China, June 1978. He is a teacher of Henan university of urban construction. In 2009 graduated from XiDian University, major in computer technology, received a master's degree in engineering.

Jian Wan, born in Hubei Huanggang in the north of China, May 1986. In 2009 graduated from central China normal university.