Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 784392, 9 pages
http://dx.doi.org/10.1155/2014/784392

*Research Article*

# Approach for Text Classification Based on the Similarity Measurement between Normal Cloud Models

**Jin Dai and Xin Liu**

*College of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*

Correspondence should be addressed to Jin Dai; daijin@cqupt.edu.cn

Received 16 October 2013; Accepted 8 January 2014; Published 23 February 2014

Academic Editors: R. Valencia-García and Y.-B. Yuan



The similarity between objects is the core research area of data mining. In order to reduce the interference of the uncertainty of nature language, a similarity measurement between normal cloud models is adopted to text classification research. On this basis, a novel text classifier based on cloud concept jumping up (CCJU-TC) is proposed. It can efficiently accomplish conversion between qualitative concept and quantitative data. Through the conversion from text set to text information table based on VSM model, the text qualitative concept, which is extraction from the same category, is jumping up as a whole category concept. According to the cloud similarity between the test text and each category concept, the test text is assigned to the most similar category. By the comparison among different text classifiers in different feature selection set, it fully proves that not only does CCJU-TC have a strong ability to adapt to the different text features, but also the classification performance is also better than the traditional classifiers.

## 1. Introduction

Text classification is key technology in text mining, which denotes the task of assigning raw text documents to one or more predefined categories. This is a direct concept from machine learning, which implies the declaration of a set of labeled categories as a way to represent the documents and a statistical classifier trained with a labeled training set. Classification is the process in which objects are initially recognized, differentiated, and understood and implies that objects are grouped into categories, usually for some specific purposes. Classification is fundamental in prediction, inference, and decision making. However, there are a variety of ways to approach classification task [1].

In recent years, with the continued development and innovation of information technology and natural language processing (NLP) fields, it has laid a solid theory and practice basis for text classification. An increasing number of supervised classification approaches have been developed for various types of classification tasks, such as decision trees (DT) [2, 3], neural networks (NN) [4, 5], naive Bayes (NB) [6–8], support vector machines (SVM) [9–12], and $k$ nearest neighbor($k$NN) [13–15]. These classifiers have their own characteristics, but in the perspective of comprehensive performance, SVM, $k$NN, and NB methods are more excellent [16].

SVM can handle high-dimensional sparse text and is not sensitive to the characteristics of relevance. But its classification speed is too slow and lacks the method for multi-class classification. In addition, the choice of kernel function and parameters still depended on experience [9]. The $k$NN method has the advantage of simple and stable performance and especially has good anti-interference to the noise data. But, it requires a large number of samples to get threshold. At the same time, how to choose $k$ value has a direct impact on the classification performance [14]. NB is simple, efficient, and stable. It requires few parameters and is not sensitive to the missing data. However, NB assumes that the attributes are mutually independent, but in practice the assumption is often not established. Especially, if the amount of attributes is larger or the correlation is great between attributes, its efficiency is low [17].

Cloud model [18–20] is an uncertainty conversion model between the quality concepts and its quantity value expressed by the natural language value. The character of cloud can be expressed by expected value (Ex), entropy (En), and hyper entropy (He) [18]. Ex is the center value of concept in the theory field, which represents the value of the qualitative

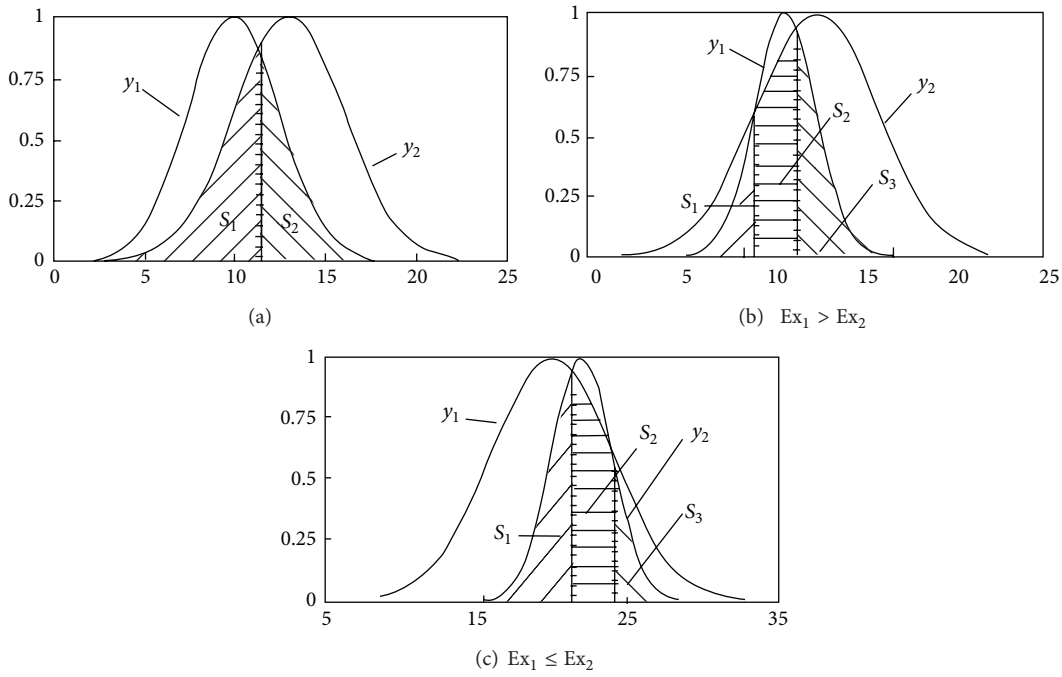



Figure 1: Similar situation between normal clouds.

concept. En is the measuring of the fuzziness of qualitative concept, which reflects the numerical range that can be accepted by this concept in the theory field. He embodies the uncertainty of the qualitative concept. The bigger the entropy is, the bigger the numerical range can be accepted by the concept and the fuzzier the concepts are.

Given three digital features Ex, En, and He, the vector $C(\mathrm{Ex}, \mathrm{En}, \mathrm{He})$ is called cloud vector, which represents a qualitative concept.

With the efficient conversion function between qualitative and quantitative data, the concept extraction method of cloud model is applied to text classification. Through the conversion from text collection to text information table based on VSM model, the text qualitative concept, which is extraction from the same category, is jumping up as a whole concept. On the basis of the cloud similarity between the test text and each category, the test text is assigned to the most similar category. By the comparison among different text classifiers based on different feature selection methods, it has not only a strong ability to adapt to the different text features, but also better classification performance than the traditional classifiers.

## 2. Similarity Measurement between Normal Cloud Models

In cloud model theory, the cloud vector $\vec{C} = \{\mathrm{Ex}, \mathrm{En}, \mathrm{He}\}$ is used to descript the qualitative concept. When making comparison of the similarity between vectors, Euclidean Distance or Cosine Angle is widely used. However, the physical meaning and the importance of each dimensionality of the cloud vector are completely different. Therefore, a novel similarity measurement between cloud vectors is required. Taking into account the normal cloud model with most universal, we propose the similarity degree between normal clouds.

Normal random distribution is the core of this similarity degree, which is obtained by calculating the area of two intersecting normal cloud vectors. The intersecting stations of two normal clouds are shown in Figure 1 (the shadow area presents each other's similarity).

Given $\vec{C}_1 = \{\mathrm{Ex}_1, \mathrm{En}_1, \mathrm{He}_1\}$, $\vec{C}_2 = \{\mathrm{Ex}_2, \mathrm{En}_2, \mathrm{He}_2\}$, $y_1(x)$ and $y_2(x)$ are,respectively, the distribution function of $\vec{C}_1$ and $\vec{C}_2$, where $x_0$ is the intersection of the two curves $y_1$ and $y_2$. Then, the intersecting area of $\vec{C}_1$ and $\vec{C}_2$ is

$$\begin{aligned} S &= \int_{-\infty}^{x_0} y_2(x)\,dx + \int_{x_0}^{\infty} y_1(x)\,dx \\ &= \int_{-\infty}^{z_2} \phi(z)\,dz + \int_{z_1}^{\infty} \phi(z)\,dz, \end{aligned} \tag{1}$$

where $z_1 = (x_0 - \mathrm{Ex}_1)/\mathrm{En}_1$, $z_2 = (x_0 - \mathrm{Ex}_2)/\mathrm{En}_2$, $\phi(z)$ is standard normal distribution. If $x_0$ is known, $z_1$ and $z_2$ are then obtained. Enquiring the table of standard normal distribution, the intersecting area $S$ can be calculated.

On the basis of the intersection between normal cloud vectors, $|z_1| = |z_2|$, then obtains that

$$\begin{aligned} x_0^{(1)} &= \frac{\mathrm{Ex}_2\mathrm{En}_1 - \mathrm{Ex}_1\mathrm{En}_2}{\mathrm{En}_1 - \mathrm{En}_2}, \\ x_0^{(2)} &= \frac{\mathrm{Ex}_1\mathrm{En}_2 + \mathrm{Ex}_2\mathrm{En}_1}{\mathrm{En}_1 + \mathrm{En}_2}. \end{aligned} \tag{2}$$

In light of "3En" principle of normal distribution, 99.74% of values are in $[\mathrm{Ex} - 3\mathrm{En}, \mathrm{Ex} + 3\mathrm{En}]$. So, when calculating the similarity of two normal distributions, we only consider the

distribution of variables in the interval. It would be well as if set $\mathrm{Ex}_1 \le \mathrm{Ex}_2$; then, there are the three following situations about the distribution of $x_0^{(1)}$ and $x_0^{(2)}$.

(1) If $x_0^{(1)}, x_0^{(1)} \notin [\mathrm{Ex}_2 - 3\mathrm{En}_2, \mathrm{Ex}_1 + 3\mathrm{En}_1]$, indicating that the values distribution of intersections can be neglected, so $S = 0$.

(2) If there is a point ($x_0^{(1)}$ or $x_0^{(2)}$) in $[\mathrm{Ex}_2 - 3\mathrm{En}_2, \mathrm{Ex}_1 + 3\mathrm{En}_1]$, as shown in Figure 1(a), then,

$$S = S_1 + S_2 = \int_{-\infty}^{z_2} \phi(x)\,dx + \int_{z_2}^{\infty} \phi(x)\,dx = \int_{-\infty}^{z_2} \phi(x)\,dx + \left(1 - \int_{-\infty}^{z_1} \phi(x)\,dx\right), \tag{3}$$

where $z_1 = (x_0 - \mathrm{Ex}_1)/\mathrm{En}_1$, $z_2 = (x_0 - \mathrm{Ex}_2)/\mathrm{En}_2$.

(3) If $x_0^{(1)}$ and $x_0^{(2)}$ are in the interval $[\mathrm{Ex}_2 - 3\mathrm{En}_2, \mathrm{Ex}_1 + 3\mathrm{En}_1]$ simultaneously, as shown in Figures 1(b) and 1(c), then,

$$S = S_1 + S_2 + S_3 = \begin{cases} \displaystyle\int_{-\infty}^{z_1^{(1)}} \phi(x)\,dx + \left(\int_{-\infty}^{z_2^{(2)}} \phi(x)\,dx - \int_{-\infty}^{z_2^{(1)}} \phi(x)\,dx\right) + \int_{z_1^{(2)}}^{\infty} \phi(x)\,dx & \mathrm{Ex}_1 > \mathrm{Ex}_2 \\ \displaystyle\int_{-\infty}^{z_2^{(1)}} \phi(x)\,dx + \left(\int_{-\infty}^{z_1^{(2)}} \phi(x)\,dx - \int_{-\infty}^{z_1^{(1)}} \phi(x)\,dx\right) + \int_{z_2^{(2)}}^{\infty} \phi(x)\,dx & \mathrm{Ex}_1 \le \mathrm{Ex}_2, \end{cases} \tag{4}$$

where $x_0^{(1)} \le x_0^{(2)}$, $z_i^{(j)} = (x_0^{(j)} - \mathrm{Ex}_i)/\mathrm{En}_i$.

Considering the normalization of the similarity, $S$ must be normalized. The area is seen as the similarity of two normal distributions after it is normalized.

*Definition 1* (similarity of normal cloud). Given normal cloud $\vec{C}_i = \{\mathrm{Ex}_i, \mathrm{En}_i, \mathrm{He}_i\}$, $\vec{C}_j = \{\mathrm{Ex}_j, \mathrm{En}_j, \mathrm{He}_j\}$, their similarity degree is

$$\gamma\left(\vec{C}_i, \vec{C}_j\right) = \frac{2S}{\left(\sqrt{2\pi}\left(\mathrm{En}_i + \mathrm{En}_j\right)\right)}, \tag{5}$$

where $S$ is the intersection area between $\vec{C}_i$ and $\vec{C}_j$.

## 3. Cloud Virtual Pan-Concept-Tree and Concept Jumping Up

*3.1. Cloud Virtual Pan-Concept-Tree.* The core idea of data mining is to discover and obtain the potential knowledge. In this process, the knowledge layers will be different with different knowledge granularity. To deal with the uncertainty of qualitative concepts, cloud model can be used to text concept representation. Text concepts set $C$ is composed of a series of basic concepts, which can be represented as cloud vector $C_i(\mathrm{Ex}_i, \mathrm{En}_i, \mathrm{He}_i)$. On this basis, text virtual concept tree can be structured layer by layer.

The virtual concept tree, which is based on cloud model, is uncertainty. On the same layer, the distinction between the various concepts is flexible. A certain degree of overlap is allowed. In other words, the attributes with the same value may belong to different concepts and different attribute has different contribution to concepts. In the construction process, the layer of concept extraction is uncertain. It can be extracted from the bottom layer or the upper layers.

In the data mining process, with the increasing of the granularity and the abstraction degree to concept, there is no physical structure of pan-concept-tree. In the same time, there is no the process of the concept climbing and jumping up layer by layer. The granularity of concepts is continuous and can jump up to any greater granularity.

Figure 2 shows a pan-concept-tree for person age distribution based on cloud model.

On the basis of cloud pan-concept-tree, a similar pan-concept-tree model for text classification is proposed (Figure 3).

*3.2. Concept Jumping Up and Merger.* The jumping up of concept refers to directly rasie the concept to the required granularity or layer with the leaf nodes in pan-concept-tree.

The strategies for concept jumping up are as follows:

(a) with the user-specified amount of concept;

(b) automatic jumping up without the amount of concept;

(c) man-machine interactive jumping up.

The concept jumping up process of pan-concept-tree usually only visits the original dataset a few times. When the original dataset is large, it can reduce the computing overhead. The strategy (a) and (b) access dataset only one time and strategy (c) requires more, whose I/O overhead depends on the number of human-machine interaction. In fact, strategy (c) can be achieved by invoking strategy (a) repeatedly.

In text classification practice, strategy (a) is mainly used to concept jumping up because the number of text categories is known. In the concept jumping up process, the merger between concepts is the most important operation. It can merge two adjacent concepts and forms an upper layer (or thicker granularity) concept. In cloud model, the concept merger operations are described as follows [7, 8].

Given two adjacent concepts, which can be signed as cloud vector $C_1(\mathrm{Ex}_1, \mathrm{En}_1, \mathrm{He}_1)$ and $C_2(\mathrm{Ex}_2, \mathrm{En}_2, \mathrm{He}_2)$, the merged cloud vector is $C(\mathrm{Ex}, \mathrm{En}, \mathrm{He})$, so

$$\mathrm{Ex} = \frac{\mathrm{Ex}_1\mathrm{En}_1 + \mathrm{Ex}_2\mathrm{En}_2}{\mathrm{En}_1 + \mathrm{En}_2};$$

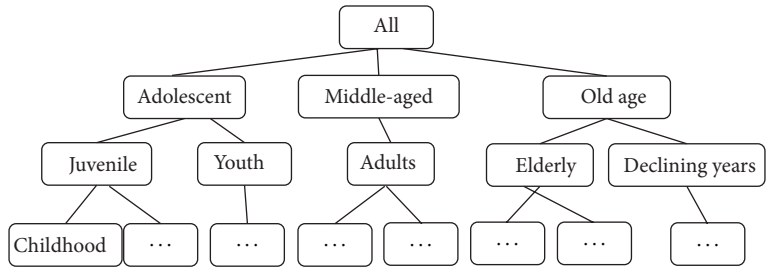



Figure 2: Cloud pan-concept-tree for person age distribution.

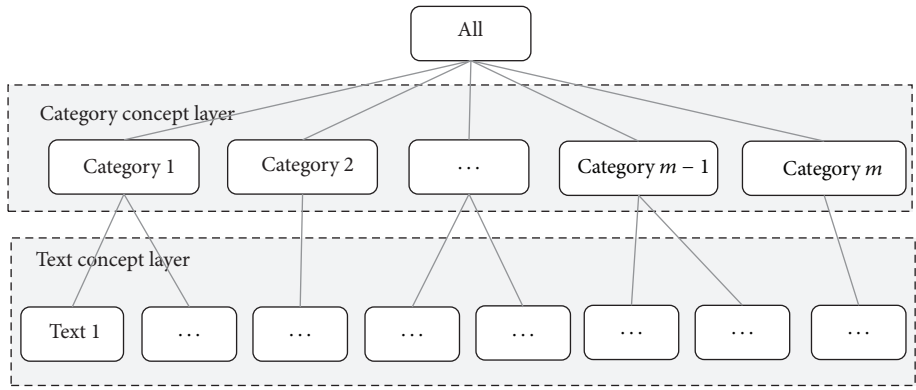



Figure 3: Pan-concept-tree for text classification.

$$\mathrm{En} = \mathrm{En}_1 + \mathrm{En}_2;$$
$$\mathrm{He} = \frac{\mathrm{He}_1\mathrm{En}_1 + \mathrm{He}_2\mathrm{En}_2}{\mathrm{En}_1 + \mathrm{En}_2}. \quad (6)$$

## 4. Text Classification Approach Based on Cloud Concept Jumping Up

In order to apply cloud model theory to text mining, we need to implement the corresponding preoperation to text, which involves the construction of text information table and conversion.

### 4.1. Text Information Table and Conversion

*Definition 2.* An information table [5] is defined as $S = \langle U, R, V, f \rangle$, where $U$ is a nonempty finite set of objects, and $R$ is a nonempty finite set of attributes, $R = C \cup D$, where $C$ is the set of condition attributes and $D$ is the set of decision attributes, $D \neq \phi$. $V = \cup V_p$, $p \in R$, and $V_p$ are the domain of the attribute $p \cdot f : U \times R \rightarrow V$ is a total function such that $f(x_i, p) \in V_p$ for every $p \in R$, $x_i \in U$.

On the basis of information table, text information table is given.

*Definition 3.* Given information table $S = \langle U, R, V, f \rangle$, where $U$ is text set, $R = C \cup D$ is a nonempty finite set of attributes, where $C$ is the set of text feature attributes and $D$ is the set of categories. $V = \cup V_p$, $p \in R$, and $V_p$ are the domain of the attribute $p \cdot f : U \times R \rightarrow V$ is a total function such that $f(x_i, p) \in V_p$ for every $p \in R$, $x_i \in U$.

Using cloud model to describe the uncertainty between texts, it must be met that the values of different attribute in the text belong to the same domain. That is, the values of different attribute should have the same physical meaning. But the values and the meaning of attributes of the existing text information table are so different that we must find one way to convert them to the same physical domain. In this paper, a text information table conversion algorithm is proposed using statistical method.

*Algorithm 4.* The conversion algorithm for text information table.

*Input.* text information table $S$
*Output.* conversion text information table $S_c$

Step 1. Set $S_c = S$

Step 2. For each $D_k$ in $D$, $Do//D$ is categories set of $S_c$

(1) $S_{jk} = \sum_{i=1}^{m} U_{ij}$, $\overline{D}_{jk} = (1/m)S_{jk}//m$ is the number of the texts whose category is $D_k$, $U_{ij}$ is the value of the $j$th attribute of the $i$th text.

(2) $U_{ij} = |U_{ij} - \overline{D}_{jk}|$ //Compute fluctuation degree of the attribute values of all sample

(3) $U_{ij} = U_{ij}/S_{jk}$ // Normalization
Next

Step 3. Return $S_c$.

After the conversion by Algorithm 4, the attributes of the text information table are changed into the same physical space. The novel table shows the fluctuation degree of the values in the different category and describes their statistical distribution.

*4.2. Cloud Concept Jumping Up Classifier.* On the basis of text information table conversion and similarity calculation, the text classifier based on cloud concept jumping up (CCJU-TC) is proposed. The classifier works as follows (Figure 4).

Whole CCJU-TC algorithm is divided into text preprocessing, text information table conversion, category concept extraction, text cloud model conceptual similarity, and several other components.

*Algorithm 5.* CCJU-TC (The text classifier based on cloud concept jumping up)

*Input.* training text set $T$, test text $d$ (unknown category)
*Output.* the category of $d$

Step 1. Text (include $T$ and $d$) segment and remove stopped items;

Step 2. Compute the weight of items by TF-IDF [9] formula;

Step 3. Text features (items) selection;

Step 4. Construct information table $S = \langle T, C \cup D, V, f \rangle$, where $C$ is text feature set and $D$ is category set;

Step 5. Invoke Algorithm 4 to convert $S$ to $S_c$

Step 6. Loop each category $D_i$ in $D$, calculate its concept cloud vector. Finally obtain categories concept set $C_{\text{concept}}$

(a) For $j = 1$ to $|D_i| - 1$

(b) $C_{T_j} = C_{T_j}$ MERGE $C_{T_{j+1}}$ //obtain category concept by concept merging, where $T_j, T_{j+1}$ are texts with the same category $D_i$, $C_{T_j}$ and $C_{T_j}$ are their corresponding cloud models.

(c) Next

(d) $C_{\text{concept}} = C_{\text{concept}} \cup C_{T_j}$;

Step 7. Compute $\max(\mu(C_d, C'_{\text{concept}p}))$, where $C_d$ is cloud vector of $d$, $\mu(C_d, C'_{\text{concept}p})$ is the cloud similarity between $C_d$ and any category concept in $C_{\text{concept}p}$. The category with most great similarity is the category of text $d$.

Step 8. Return.

## 5. Experiments and Evaluations

*5.1. Experiment on Different Datasets.* In order to evaluate the performance of CCJU-TC facility, we have acquired four datasets of varying characteristics. Each dataset has its own unique characteristics in terms of the degree of similarity between categories and the dimensionality of categories, as shown in Table 1.

Table 1: Datasets and their characteristics.

| Datasets | Degree of similarity between categories | Number of articles | Dimensionality of categories |
|---|---|---|---|
| Featured Articles (Wikipedia) | Normal | 1159 | 23 |
| Vehicles (Wikipedia) | Low | 640 | 4 |
| Mathematics (Arxiv.org) | High | 320 | 8 |
| 20-Newsgroups (CMU Text Learning Group) | Normal | 20,000 | 20 |

The Featured Articles dataset was designed and organized by our research group by extracting different types of articles from Wikipedia website. A total of 1159 articles were acquired from twenty-three randomly selected categories. 10 documents from each category have been randomly selected to build the training set. The remaining documents were utilized for testing purposes. In other word, the training set of this dataset consists of 230 documents, while the testing set consists of a total of 929 documents.

The Vehicles dataset was built by extracting vehicle related articles from Wikipedia website. This dataset was acquired by extracting articles from four subcategories in the category of "Vehicles." All the four categories are easily differentiated and each category has its own unique keywords. This dataset consists of 640 documents. Each category consists of 160 documents where 50 documents were used to build the training set and the remaining 110 documents were utilized for testing purposes. In other words, the training set consists of a total of 200 documents, while the testing set contains a total of 440 documents.

A dataset containing articles about mathematical topics has been acquired from arxiv.org. This dataset consists of eight categories regarding mathematical topics. 40 documents for each category have been collected, and the entire dataset consists of a total of 320 documents. 10 documents from each category were extracted randomly to build the training set, while the remaining 30 documents from each category were used for testing purposes.

20-Newsgroups dataset is one of the standard benchmark datasets used by many text classification research groups to evaluate the performance of their presented classification approaches. 20-Newsgroups dataset is a collection of 20,000 Usenet articles from twenty different newsgroups with 1000 articles per newsgroup. 20-Newsgroups collection has become a widely used dataset for experiments in text applications of machine learning techniques, such as text classification and text clustering. 20-Newsgroups dataset used in our experiments was acquired from the CMU Text Learning Group's website. In our experiments using this dataset, every category was divided into two subsets. 300 documents from each category were divided for training, while the remaining 700 documents were used for testing purposes.

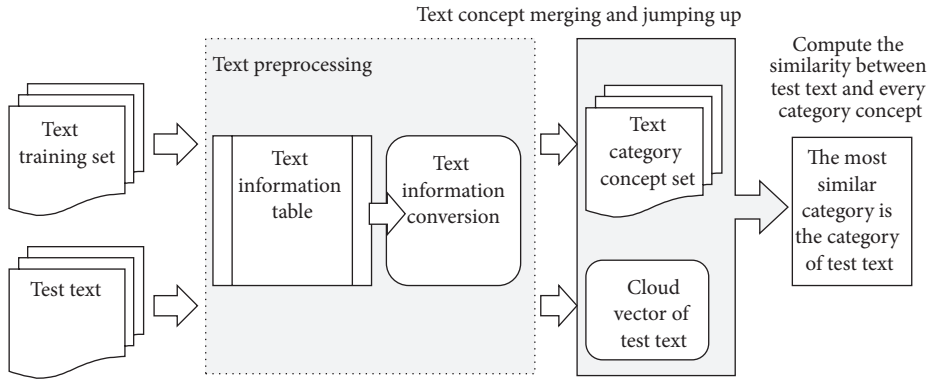



Figure 4: Text classification process based on cloud concept jumping up.

Table 2: Classification performance of different classifier with IG.

| Datasets | SVM | | | *k*NN | | | NB | | | CCJU-TC | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ |
| Featured Articles | 72 | 75 | 73.47 | 75 | 77 | 75.99 | 70 | 71 | 70.50 | 89 | 92 | **90.48** |
| Vehicles | 78 | 81 | 79.47 | 81 | 82 | 81.50 | 72 | 73 | 72.50 | 86 | 89 | **87.47** |
| Mathematics | 80 | 81 | 80.50 | 78 | 80 | 78.99 | 73 | 75 | 73.99 | 87 | 88 | **87.50** |
| 20-Newsgroups | 75 | 76 | 75.50 | 77 | 78 | 77.50 | 70 | 73 | 71.47 | 81 | 83 | **81.99** |
| Average | 76.25 | 78.25 | 77.23 | 77.75 | 79.25 | 78.49 | 71.25 | 73 | 72.11 | 85.75 | 88 | **86.86** |

Table 3: Classification performance of different classifier with CHI.

| Datasets | SVM | | | *k*NN | | | NB | | | CCJU-TC | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ |
| Featured Articles | 70 | 71 | 70.50 | 71 | 72 | 71.50 | 68 | 68 | 68.00 | 83 | 85 | **83.99** |
| Vehicles | 72 | 74 | 72.99 | 75 | 77 | 75.99 | 70 | 71 | 70.50 | 80 | 82 | **80.99** |
| Mathematics | 74 | 74 | 74.00 | 73 | 73 | 73.00 | 66 | 69 | 67.47 | 81 | 82 | **81.50** |
| 20-Newsgroups | 69 | 70 | 69.50 | 70 | 72 | 70.99 | 62 | 65 | 63.46 | 75 | 78 | **76.47** |
| Average | 71.25 | 72.25 | 71.75 | 72.25 | 73.5 | 72.87 | 66.5 | 68.25 | 67.36 | 79.75 | 81.75 | **80.74** |

In other words, the training set consists of 6000 documents and the remaining 14,000 documents are used for testing purpose.

Many research works in text document classification apply preprocesses such as stop word elimination, word stemming, and feature selection to the datasets used in their experiments and evaluations in order to obtain better experimental results.

As our research goal in this paper is to evaluate the performance of CCJU-TC facility without sacrificing the simplicity and low cost classification algorithms, we did not perform any preprocess such as stop word elimination, word stemming to the datasets that we used in our experiments. In order to reduce the interference of different feature set, IG, CHI [21, 22], and FAS [23, 24] feature selection methods are adopted.

In all experiments, we employ SVM (Torch), *k*NN ($k$ = 30), and NB classifiers as a comparison. In the way to construct the training set and test set, we apply the 3-fold cross validation, which randomly divides the text sets into three parts. Two parts are used as training sets; the other left is test set. Finally, the average of three classification results is the experimental result, which is evaluated by precision rate ($p$), recall rate ($r$), and $F_1(2 \times p \times r/(p + r))$. The details of experimental result are in Tables 2, 3, and 4.

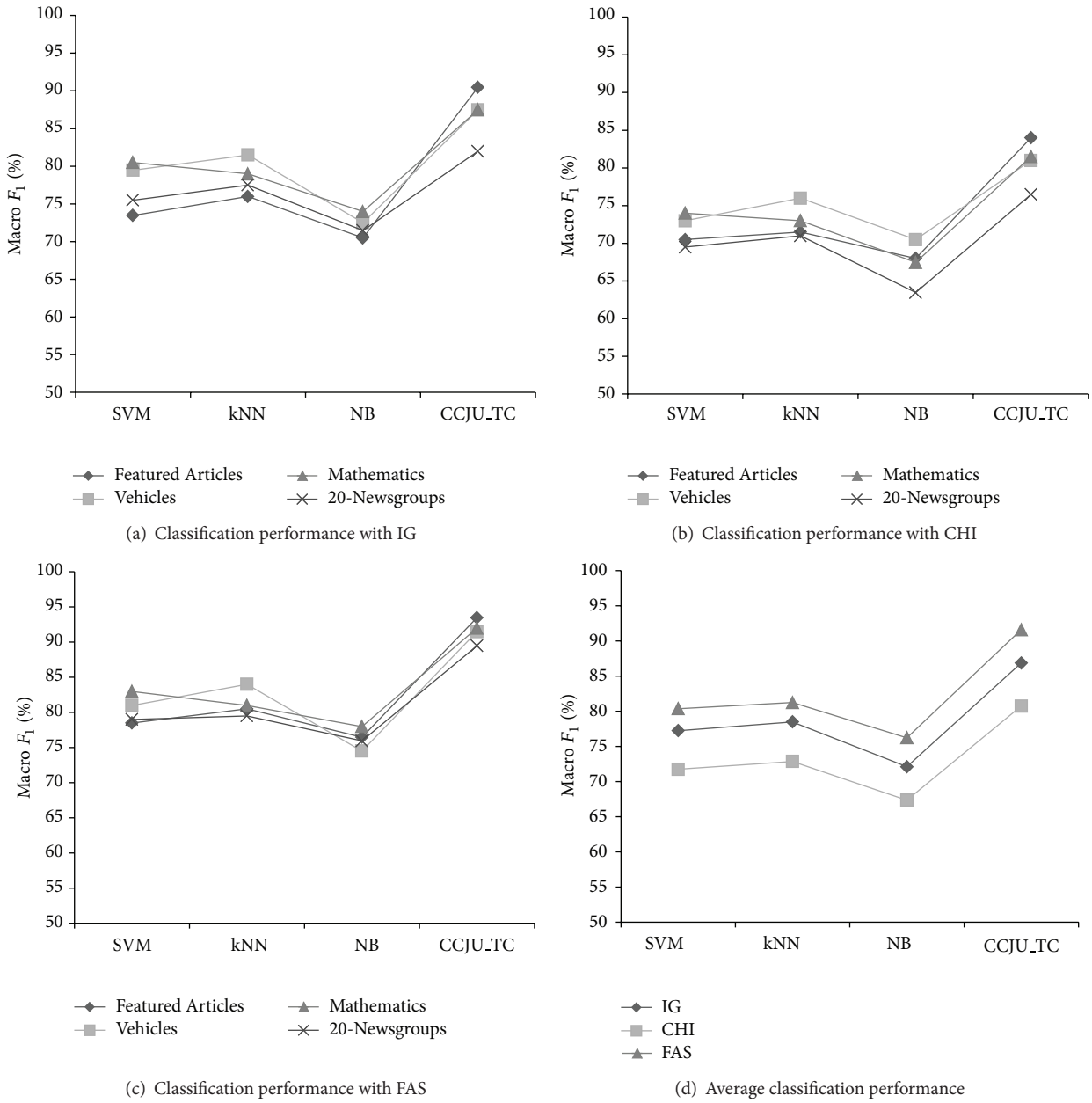
In order to compare the performance of the classifier more intuitive, Figure 5 is created based on the above tables.

*5.2. Summary of Experiment.* In the above classification experiment, CCJU-TC has the better performance, followed by *k*NN and SVM, NB classification performance is the worst (Figure 5). Meanwhile, the different feature selection method for text categorization performance impact is also different. The performance of the text classifier, which selected features by FAS, is higher and stable (Figure 5(c)). CHI feature selection method for text classification more significant impact, mainly due to over-reliance on low-frequency features, especially in the 20-Newsgroups dataset (Figure 5(b)).

Through the comparison test among multiple text classifiers with a variety of feature selection methods on the different datasets, CCJU-TC has shown an excellent text classification capability. It not only has good ability to adapt

Table 4: Classification performance of different classifier with FAS.

| Datasets | SVM | | | *k*NN | | | NB | | | CCJU-TC | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ | $p$ (%) | $r$ (%) | $F_1$ |
| Featured Articles | 78 | 79 | 78.50 | 80 | 81 | 80.50 | 75 | 78 | 76.47 | 92 | 95 | **93.48** |
| Vehicles | 80 | 82 | 80.99 | 83 | 85 | 83.99 | 74 | 75 | 74.50 | 90 | 93 | **91.48** |
| Mathematics | 82 | 84 | 82.99 | 80 | 82 | 80.99 | 77 | 79 | 77.99 | 91 | 93 | **91.99** |
| 20-Newsgroups | 79 | 79 | 79.00 | 79 | 80 | 79.50 | 74 | 78 | 75.95 | 88 | 91 | **89.47** |
| Average | 79.75 | 81 | 80.37 | 80.5 | 82 | 81.24 | 75 | 77.5 | 76.23 | 90.25 | 93 | **91.60** |




(a) Classification performance with IG

(b) Classification performance with CHI

(c) Classification performance with FAS

(d) Average classification performance

Figure 5: Classification results comparison.

different feature set, but also has better classification ability. It full proves that CCJU-TC is an efficient classifier.

## 6. Conclusion

In the present research to text mining (TM), traditional data mining methods still dominated. However, with further research, it faces more severe challenges. These difficulties, such as the huge dimensions and sparsity of text object, the high complexity of algorithm, and the requirement of prior knowledge, have seriously hampered the development of TM.

Through in-depth analysis, these problems in TM process are due to the uncertainty of natural language. The uncertainty of natural language (especially text) comes from the uncertainty of the human thinking in essence. Although it strengthens understanding of spatial and cognitive for people, brings a series of problems to TM. Therefore, from the point of reducing the complexity of natural language, if we can carry out the advanced innovation, which is based on making full use of these existing technologies, and find out a novel uncertainty artificial intelligence approach for TM, it will greatly facilitate the rapid development of TM.

CCJU-TC classifier is a novel attempt to apply uncertain knowledge acquisition tool (cloud model) to text classification in TM. The experimental result shows that it is an excellent solution. Future, more data mining methods, based on cloud model, will be carried out.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (NSFC) under Grant no. 61309014; the Natural Science Foundation Project of CQ CSTC under Grant no. cstc2013jcyjA40009, no. cstc2013jcyjA40063; and the Natural Science Foundation Project of CQUPT under Grant no. A2012-96.

## References

[1] F. Sebastiani, "Machine learning in automated text categorization," *ACM Computing Surveys*, vol. 34, no. 1, pp. 1–47, 2002.

[2] S. K. Murthy, "Automatic construction of decision trees from data: a multi-disciplinary survey," *Data Mining and Knowledge Discovery*, vol. 2, no. 4, pp. 345–389, 1998.

[3] F. De Comité, R. Gilleron, and M. Tommasi, "Learning multi-label alternating decision trees from texts and data," in *Machine Learning and Data Mining in Pattern Recognition*, pp. 35–49, Springer, Berlin, Germany, 2003.

[4] B. Yu, Z.-B. Xu, and C.-H. Li, "Latent semantic analysis for text categorization using neural network," *Knowledge-Based Systems*, vol. 21, no. 8, pp. 900–904, 2008.

[5] M. E. Ruiz and P. Srinivasan, "Hierarchical text categorization using neural networks," *Information Retrieval*, vol. 5, no. 1, pp. 87–118, 2002.

[6] A. McCallum and K. A. Nigam, "comparison of event models for naive bayes text classification," in *Workshop on Learning for Text Categorization (AAAI '98)*, vol. 752, pp. 41–48, 1998.

[7] A. Z. Broder, M. Fontoura, E. Gabrilovich, A. Joshi, V. Josifovski, and T. Zhang, "Robust classification of rare queries using web knowledge," in *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '07)*, pp. 231–238, July 2007.

[8] L. H. Lee, D. Isa, W. O. Choo, and W. Y. Chue, "High Relevance Keyword Extraction facility for Bayesian text classification on different domains of varying characteristic," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1147–1155, 2012.

[9] T. Joachims, *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*, Springer, Berlin, Germany, 1998.

[10] T.-Y. Wang and H.-M. Chiang, "One-against-one fuzzy support vector machine classifier: an approach to text categorization," *Expert Systems with Applications*, vol. 36, no. 6, pp. 10030–10034, 2009.

[11] M. A. Kumar and M. Gopal, "A comparison study on multiple binary-class SVM methods for unilabel text categorization," *Pattern Recognition Letters*, vol. 31, no. 11, pp. 1437–1444, 2010.

[12] A. Sun, E.-P. Lim, and Y. Liu, "On strategies for imbalanced text classification using SVM: a comparative study," *Decision Support Systems*, vol. 48, no. 1, pp. 191–201, 2009.

[13] Y. -Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 42–49, ACM, 1999.

[14] E. H. S. Han, G. Karypis, and V. Kumar, *Text Categorization Using Weight Adjusted K-Nearest Neighbor Classification*, Springer, Berlin, Germany, 2001.

[15] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1503–1509, 2012.

[16] J.-S. Su, B.-F. Zhang, and X. Xu, "Advances in machine learning based text categorization," *Journal of Software*, vol. 17, no. 9, pp. 1848–1859, 2006.

[17] L. H. Lee, D. Isa, W. O. Choo, and W. Y. Chue, "High Relevance Keyword Extraction facility for Bayesian text classification on different domains of varying characteristic," *Expert Systems with Applications*, vol. 39, no. 1, pp. 1147–1155, 2012.

[18] D. Y. Li, *Artificial Intelligence with Uncertainty*, National Defense Industry Press, Beijing, China, 2005.

[19] D. Y. Li and C. Y. Liu, "Study on the universality of the normal cloud model," *Engineering Science*, vol. 6, no. 8, pp. 28–34, 2004.

[20] D.-Y. Li, C.-Y. Liu, Y. Du, and X. Han, "Artificial intelligence with uncertainty," *Journal of Software*, vol. 15, no. 11, pp. 1583–1594, 2004.

[21] P. Soucy and G. W. Mineau, "Feature selection strategies for text categorization," in *Proceedings of the 16th Conferenceof the Canadian Society for Computational Studies of Intelligence (CSCSI '03)*, Y. Xiang and B. Chaib-Draa, Eds., pp. 505–509, Springer, Halifax, UK, 2003.

[22] G.-W. Zhang, D.-Y. Li, P. Li, J.-C. Kang, and G.-S. Chen, "Collaborative filtering recommendation algorithm based on cloud model," *Journal of Software*, vol. 18, no. 10, pp. 2403–2411, 2007.

[23] J. Dai and F. Hu, "Research and application of text classification based on incomplete information system," *Journal of Chongqing University of Posts and Telecommunications*, vol. 18, no. 3, pp. 397–401, 2006.

[24] J. Dai, Z. He, and F. Hu, "A high performance algorithm for text feature automatic selection based on cloud model," *Journal of Computational Information Systems*, vol. 5, no. 6, pp. 1561–1568, 2009.