Rough Sets and Confidence Attribute Bagging for Chinese Architectural Document Categorization

Xiang Zhang, Changhua Li, Lili Dong, Na Ye
School of Information and Control Engineering, Xi'an University of Architecture and Technology,
Xi'an 710055, China
Email: Zhangxiang1001@126.com, lichangh@pub.xaonline.com,donglilixjd@163.com,yenanaye@126.com

***Abstract*— Aiming at the problems of the traditional feature selection methods that threshold filtering loses a lot of effective architectural information and the shortcoming of Bagging algorithm that weaker classifiers of Bagging have the same weights to improve the performance of Chinese architectural document categorization, a new algorithm based on Rough set and Confidence Attribute Bagging is proposed for Chinese architectural document categorization. Rough sets is used to feature selection. First the cores of attributes are found by discernibility matrix and one of the cores is regarded as the start point. Then attributes' significance and dependency are used as the heuristic information to do feature selection. A Chinese architectural document classifier is designed by Confidence Attribute Bagging algorithm. The voting weights of weaker classifiers are gained by their result and the stronger classifier result is attained by weaker classifiers voting. The algorithm is applied in Attribute Bagging algorithm to design a classifier. The experimental results show that the novel method is not only easy to implement but can effectively reduce the dimensional space, and improve the accuracy of classification.**

***Index Terms*—Rough sets, feature selection, Confidence Attibute Bagging, Chinese architectural document categorization**

I. INTRODUCTION

In recent years there are tremendous growths in volumes of Chinese architecture documents available on the internet. Automatic methods for organizing the data of architecture are needed. Text categorization (TC—a.k.a. text classification, or topic spotting), is the activity of labeling natural language texts with thematic categories from a predefined set [1].

In the 90s, machine learning which has taken the place of knowledge engineering approach became main technology to text categorization. Ensemble learning has become a hot topic in machine learning studies, because it can effectively improve the generalization performance which is a basic problem of machine learning. [2]In text categorization, ensemble learning is an effective method to improve the result of text classifiers [3].

The Bagging algorithm is one of the ensemble learning algorithms and its weaker classifier has the same weight. It is unreasonable for only using category as the weight for voting, because the result of higher creditable classifier is more plausible than that of lower creditable classifier. Even if results of two text classifiers are same, one classifier has higher creditability and anther has lower creditability, they should not play the same role when they are voted.

In this paper, a confidence voting method is applied to Attribute Bagging through which a text classifier is built for Chinese architectural document. The confidence of weaker text classifiers are gained through its result and the weights of voting are obtained by confidence. The algorithm is applied in Attribute Bagging algorithm to design a Chinese text classifier. kNN algorithm is used as the weaker classifier model to classify Chinese architectural document. The result of experiment shows that this algorithm performs better than Attribute Bagging with more accuracy.

The rest of this paper is organized as follows: In section 2, we will discuss the related work. A brief introduction to the rough sets is in section 3. The new text categorization algorithm based on a confidence voting for Chinese architectural document is discussed in section 4. In section 5, we present the experiment result and analysis. Some conclusions and ideas for further research are described finally.

II. RELATED WORK

The process of text categorization could be divided into five steps. The first step is text preprocessing and document indexing is the second step. The third step is feature selection. Building text classifier is the fourth step. Finally performance of classifier is measured. The process of text colorization is shown in figure 1.

A. Preprocessing

The preprocessing step is to transform documents which typically are strings of characters into a representation suitable for computer processing. The main task of this step is that HTML or other tags and stop words are removed. The stop words are frequent words that carry no information.

B. Document indexing

In text categorization, each document is usually represented by a vector of n weighted index terms. This is often referred to as the bag of words approach to document representation. Vector space model is most commonly document representation and its weights usually range between 0 and 1.

doi:10.4304/jsw.6.4.628-635

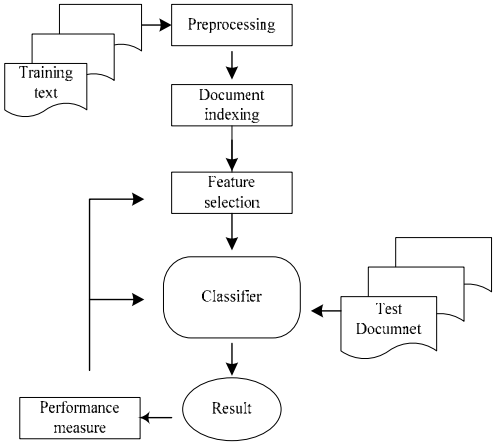

Figure 1. The process of text categorization

Boolean weighting, the simplest approach, is to let the weight be 1 if the word occurs in the document and 0 otherwise.

Word frequency weighting is to use the frequency of the word in the document.

The previous two approaches do not take into account the frequency of the word throughout all documents in the training sets. The tf*idf weighting is a well known approach for computing word weights which will introduced in section 4.

C. Feature selection

Text categorization based on machine learning is essentially statistical pattern recognition. The problem of feature selection is an important step of the statistical pattern recognition.

Feature selection is often used to reduce the size of feature dataset from $|\tau|$ to $|\tau'| << |\tau|$; the set τ' is called the reduced feature set. There are a variety of feature selection methods for text categorization, Information Gain, χ^2, Document Frequency, Bi-Normal Separation and Odds Ratio were reported to be most effective[1]. The above feature selection techniques are then defined as follows:

1. Document frequency: Features are selected by frequency in document, with a threshold.

2. Information gain measurement: Given a set of categories $|\vartheta = \{c_i\}_{i=1}^m|$, the information gain of term x is given by:

$$IG(x) = -\sum_{i=1}^{m} P(c_i)\log P(c_i) + P(x)\sum_{i=1}^{m} P(c_i|x) + P(\bar{x})\sum_{i=1}^{m} P(c_i|\bar{x})\log P(c_i|\bar{x}) \quad (1)$$

3. Mutual information measurement: Mutual information for a term in class c is given by

$$MI(x,c) = \log\frac{P(x \wedge c)}{P(x)\cdot p(c_i)} \quad (2)$$

For a set of categories $|\vartheta = \{c_i\}_{i=1}^m|$ .Mutual information of each term t is calculated by

$$MI(x,\vartheta) = \sum_{i=1}^{m}\log\frac{P(x \wedge c_i)}{P(x)\cdot P(c_i)} \quad (3)$$

4. χ^2 (CHI):

$$N \cdot \frac{P(t_k,c_i)P(\bar{t_k},\bar{c_i}) - P(t_k,\bar{c_i})P(\bar{t_k},c_i)}{P(t_k)P(\bar{t_k}) - P(c_i)P(\bar{c_i})} \quad (4)$$

5. Bi-Normal Separation (BNS):

$$|F^{-1}(P(t_k|c_i)) - F^{-1}(P(t_k|\bar{c_i}))| \quad (5)$$

Where F is the cumulative probability function of the standard Normal distribution.

6. Odds Ratio:

$$\frac{P(t_k|c_i)\cdot P(t_k|\bar{c_k})}{(1-P(t_k|c_i))\cdot P(t_k|\bar{c_k})} \quad (6)$$

However, the above methods are based on conditional independent assumption. Conditional independent assumption only considers the relevant feature between the term and the category, but neglects of the relevant the feature between the terms. Because of this, there are many redundancies in feature subset.

D. text categorization classifier

There are many text categorization classifiers such as naïve Bayes, Decision Trees, K-nearest neighbors (kNN) algorithm, Support Vector Machines (SVMs) and Ensemble learning [1].

1) Rocchio's algorithm

Rocchio's algorithm is a classic algorithm in information retrieval. In this approach, a prototype vector of each class c_i is built. Distance between document vector $\vec{d}$ and the prototype vector is computed by the dot product or by the Jaccard similarity measure. In this way document vector $\vec{d}$ is classified. This method is learned very quickly.

2) The naïve Bayes classifier

The naïve Bayes classifier is that a test text probability of each class is estimate by Bayes theorem.

$$p(c_j|d) = \frac{p(c_j)|p(d|c_j)}{p(d)} \quad (7)$$

We suppose that the words are conditionally independent in the given document. Therefore The Bayes theorem equation can be simplified as follow:

$$p(c_j|d) = p(c_j)\prod_{i=1}^{M}|p(d|c_j) \quad (8)$$

$$p(c_j) = \frac{N_j}{N} \quad (9)$$

Where N_j is the number of document from class c_j and N is the total number of documents in training set.

$$p(d|c_j) = \frac{1 + N_{ij}}{M + \sum_{k=1}^{M} N_{kj}} \quad (10)$$

where N_{ij} is the number of times word i occurred within documents from class c_j in the training set.

The denominator of fraction is the total number of words in documents from class c_j.

The Naive Bayes classier is very effective although the fact that the assumption of conditional independence is generally not true for word occurred in documents.

3) decision tree algorithm

A decision tree (DT) is a tree in which internal nodes are labeled by terms, branches departing from them are labeled by tests on the weight that the term has in the test document, and leafs are labeled by categories. Such a classifier categorizes a test document $\overrightarrow{d_j}$ by recursively testing for the weights that the terms labeling the internal nodes have in vector, until a leaf node is reached; the label of this node is then assigned to d_j. Most such classifiers use binary document representations, and thus consist of binary trees [1].

4) kNN algorithm

kNN is a example-based method that is lazy learning. To classify an unknown document vector $\vec{d}$, kNN looks at whether the k training documents most similar to $\vec{d}$ also are in c_i, if the answer is positive for a large enough proportion of them, a positive decision is taken, and a negative decision is taken otherwise. Similarity may be measured by for example the Euclidean distance or the cosine between two document vectors.

5) support vector machine

The support vector machine (SVM) method may be seen as the attempt to find decision surfaces that separate the positive from the negative training examples. The decision surface that separates the positives from the negatives by the widest possible margin, i.e. SVMs were usually conceived for binary classification problems, and only recently have they been adapted to multiclass classification [1].

6) Bagging algorithm

Ensemble learning combines outputs from multiple classifiers, and it is one of the most important techniques for improving classification accuracy in machine learning. Bagging is the most popular method of ensemble learning [4]. In this paper, Bagging algorithm is used to design the text classifier. Therefore, we only introduce Bagging algorithm in this section.

In Bagging, k training sets are produced by randomly sampled k times with replacement in the original training set. So some training instances are never in the new training sets and some are repeated in new training sets. The k weaker classifiers are trained by k new training sets, which can be used to classify new text. The result of classification is obtained by equal weight voting on all k weaker classifiers. Voting gives a significant improvement in classification accuracy and stability [5].

Algorithm1: Bagging algorithm

Input: training set S, machine learning algorithm C4.5, integer T (number of iteration);

Output: result of classifier H^*;

Step1: For i =1 to T;

Step 1.1 $S_i = resampling\ (S)$ //get the training subset S_i by re-sampling S;

Step1.2 $H_i = C4.5(S_i)$ //C4.5is applied to S_i;

Step 2 $H^*(x) = \underset{y \in Y}{\operatorname{argmax}} \sum_{i=1}^{T} I(H_i(x) = y)$ // the result H^* is gotten by vote;

Step 3 return the result H^*.

E. Performance Measures

How to measure the performance of the classifiers is an important problem in text categorization. Many measures have been used such as recall, precision, F-measure, micro-averaging, macro-averaging and break-even point etc, each of which has been designed to evaluate some aspect of the categorization performance of a system.

Recall, precision and F-measure method will introduced in section 5.in this section we introduce micro-averaging and macro-averaging.

When dealing with multiple classes, there are two conventional methods of averaging these measures (i.e. recall, precision, F1-measure) namely macro-averaging and micro-averaging.

TABLE I. THE UTILITY MATRIX 1

Category set C_i		**Expert judgments**	
		YES	**NO**
Classifier Judgments	**YES**	**a**	**b**
	NO	**c**	**d**

TABLE II. THE UTILITY MATRIX 2

Category set $\dot{C} = \{c_1, c_2, ..., c|c|\}$		**Expert judgments**	
		YES	**NO**
Classifier Judgments	**YES**	$A = \sum_{i=1}^{|c|} a$	$B = \sum_{i=1}^{|c|} b$
	NO	$C = \sum_{i=1}^{|c|} c$	$D = \sum_{i=1}^{|c|} d$

Micro-average performance scores are determined as follows:

$$\overline{Mi}\,\mathrm{Re} = \frac{A}{M} = \frac{A}{A + C} = \frac{\sum_{i=1}^{|c|} a}{\sum_{i=1}^{|c|} a + \sum_{i=1}^{|c|} c} \quad (11)$$

$$\overline{Mi}\,\mathrm{Pr}e = \frac{A}{A + B} = \frac{\sum_{i=1}^{|c|} a}{\sum_{i=1}^{|c|} a + \sum_{i=1}^{|c|} b} \quad (12)$$

Macro-average performance scores are determined as follows:

$$\overline{Ma}\,\mathrm{Re} = \frac{\sum_{i-1}^{|c|} re}{|C|} \quad (13)$$

$$\overline{Ma}\,\mathrm{Pr}\,e = \frac{\sum_{i=1}^{|c|} \mathrm{Pr}\,e}{|C|} \quad (14)$$

There is an important distinction between the two types of averaging: micro-averaging gives equal weight to every document while macro-averaging gives equal weight to each category [6].

III. ROUGH SETS THEORY

In this section we provide some basic definitions of rough set theory. In 1982, Pawlak introduced the theory of rough sets. The rough sets theory has been developed for knowledge discovery in database and experimental data sets. In rough sets theory, the data is organized in a table called as decision table, which consists of objects and Attributes. Rows of the decision table correspond to objects and columns correspond to Attributes. In the data set, a class label indicates the class to which each row belongs. The class label is called decision Attributes [7, 8, 9].

Information system is defined as a 4-tuple $S=<U,A,V,f>$, Where U is a finite, non-empty set, called the universe; A is a non-empty finite set of attributes. $V = U_{a\in A} V_a$ is a domain of attribute a, and $f := U \times A \to V$ is called an information function such that $f(x,a) \in V_a$,for $\forall a \in A, \forall x \in U$.

Decision table is defined as follows:

An information system $S =< U, A, V, f >$, where $A = C \cup D$ and $C \cap D = \phi$,where C is a set of condition attributes and D is a set of decision attributes.

Indiscernibility relation is defined as that let $S =< U, A, V, f >$ be an information system, every $P \subseteq A$ generates an indiscernibility relation IND(P) on U, which is defined as follows:

$$IND(P) = \{(x,y) \in U \times U : f(x,a) = f(y,a), \forall a \in P\} \quad (15)$$

$U / IND(P) = \{C_1, C_2, ...C_k\}$ is a partition of U by P, every C_i is an equivalent class, $\forall x \in U$,the equivalent class of x in relation $U / IND(P)$ is defined as follows:

$$[x]_{IND(P)} = \{y \in U : f(y,a) = f(x,a), \forall a \in p\} \quad (16)$$

Lower approximation is defined as follows

Let $X \subseteq U$ represents a vague concept, and then the lower approximation of the indiscernibility relation can be defined as equation (9)

$$R_{-}(X) = \cup\{X_i \in U / I \mid X_i \subseteq X\} \quad (17)$$

Upper approximation is defined as follows

Let $X \subseteq U$ represents a vague concept, and then the upper approximation of the indiscernibility relation can be defined as equation (10)

$$R^{-}(X) = \cup\{X_i \in U / I \mid X_i \subseteq X\} \quad (18)$$

Attribute reduction is defined as follows:

Let $S = \langle U, C \cup D, V, f \rangle$ be a decision table, the set of attributes $P (P \subseteq C)$ is a reduction of attributes C, which satisfies the following conditions:

$$\gamma_P(D) = \gamma_C(D) and \gamma_P(D) \neq \gamma_{P'}(D), \forall P' \subset P \quad (19)$$

A reduction of condition attributes C is a subset that can discern decision classes with the same discriminating capability as C, and none of the attributes in the reduction can be eliminated without decreasing its discriminating capability.

The **discernibility matrix** is a symmetric $|U| \bullet |U|$ matrix which can capture the discrimination information involved all conditional attributes in an information system. Its entries c_{ij} can be defined as follows:

$$\{a \in A / a(x_i) \neq a(x_j)\} \quad (20)$$

If $d(x_j) \neq d(x_i)$, ϕ otherwise.

Attribute significance is defined as follows:

M is a discernibility matrix. C is condition attributes and D is decision attributes. if $R \subset C$ and $\forall a \in C - R$, significance of attribute a is

$$SGF(a,R,D) = p(a) \quad (21)$$

Where $p(a)$ is a function of attribute frequency, which is defined as the number times attribute a occurred in M.

Attribute dependency $\gamma_R(P)$ which attribute set P depends on is defined as follows:

$$\gamma_R(P) = \frac{card(POS_R(P))}{card(U)} \quad (22)$$

Where $card(POS_R(P))$ is the cardinality of the positive region and $card(U)$ is the cardinality of the universe.

IV. CHINESE ARCHITECTURAL DOCUMENT CATEGORIZATION ALGORITHM

A. Text preprocessing

The first step is a series of text preprocessing to transform Chinese architectural documents, which typically are string of character, into vectors suitable for the machine learning algorithm and the classification task. The perhaps most commonly used document representation is the so-called VSM (vector space model) [1]. Preprocessing steps are as follows:

In the case of Chinese architectural document, there is no obvious space between Chinese words. Chinese words segmentation is to divide particular text into some words with uncertain lengths. Then the lowest frequency words are removed. Finally the document can be represented by the bag of words .The document D can be converted to $d = ((f_1, w_1), (f_2, w_2), ..., (f_i, w_i))$, where each f_i is a document word, and w_i denotes its frequency.

In VSM, Chinese architecture documents are represented by vectors of words. Usually, documents are represented by a word-by-document matrix A, where each entry represents the occurrence of a word in a document, i.e.,

$$A = (a_{ik}) \quad (23)$$

Where a_{ik} is the weight of word i in document k. There are several ways of determining the weight a_{ik} such as Boolean weighting and word frequency weighting etc. The *tfidf* weighting function is simple but performs well in many situations. In this paper, we use this method in our study and it is defined as:

$$tfidf(t_k, d_j) = \#(t_k, d_j).\log\frac{|T_r|}{\#_{Tr}(t_k)} \quad (24)$$

Where $\#(t_k, d_j)$ denotes the number of time t_k occurs in d_j, $\#_{Tr}(t_k)$ denotes the document frequency of term t_k ,that is, the number of documents in T_r in which t_k occurs[1]. In order to make weights fall in the [0, 1] interval and documents be represented by vectors with equal length, the weights resulting from *tfidf* are often normalized by cosine normalization, given by:

$$w_{jk} = \frac{tfidf(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|}(tfidf(t_s, d_j))^2}} \quad (25)$$

B. Feature selection

1) Feature weight discretization

Rough sets cannot directly deal with continuous features, so weights of those features must be discrete. In this paper Equal Interval width algorithm [4] is used to divide the continuous features. According to parameters given by the user, the continuous value of the feature weight is divided into equal interval width. Therefore the continuous feature is to be discrete and normalized.

2) Construct decision table

The universe U consists of all the training text sets of each text class. Terms of text are regarded as attribute sets, namely condition attributes are represented by words (or term, feature); the decision attributes indicate the class label. Decision table of training phase is denoted as table Ⅲ.

TABLE III. DECISION TABLE

U	C_1	C_2	…	C_h	Class
x_1	4	1	…	3	d_1
……	……	……	…	……	……
x_m	8	4	…	0	d_n

Where x_i represents training text; d_i indicates the class label of x_i; c_i is the attribute sets of document x_i; the value of c_i is discrete weight of the feature. If no word occurs in a Chinese architectural document, then the weight of the feature is 0.

3) Feature selection

In this paper, a feature selection method based on rough set for Chinese architectural document is put forward. In this method, firstly the cores of the attributes are found through the discernibility matrix. Secondly one of the cores is regarded as the start point of reduction process. Then according to the significance of attribute, the attributes are put into reduction feature form the more significant to less significant. If there are two attributes whose significances are equal, the more reliant attribute is firstly put into reduction feature set. Finally the feature selection ends until the discernibility matrix is null.

Rough sets for Chinese architectural document feature selection is described as follows:

Algorithm2. Rough sets for feature selection

Input: a decision table $T = \langle U, C \cup D, V, f \rangle$, $U = \{x_1, x_2, ...x_m\}$, $C = \{c_1, c_2, ...c_n\}$

Output: a reduction of T denoted as R.

Step 1: construct the discernibility matrix M of T, c_{ij} is the elements of the M. element of c_{ij} is the different attribute sets of x_i and x_j $C_{ij} \subseteq C$, $c_k \in C_{ij}$;

Step 2: delete the rows in the M which are all 0s.

Step3: calculate the attribute core sets C_O of M, $R = C_O$

Step4: while ($M \neq \phi$)

Step 4.1 $Q = \{C_{ij} : C_{ij} \cap R \neq \phi, i \neq j, i, j = 1,2,3,...n\}$, $M = M - Q$, $B = C - R$;

Step 4.2 $\forall c_k \in B$, calculate the value of $SGF(c_k, R, D)$

Step 4.3 choose the attribute a which $SGF(c_k, R, D)$ is the max.(choose the attribute which Significance is max), $R \leftarrow R \cup \{a\}$

Step 4.4 if there are two attribute a_p and a_q with same $SGF(c_k, R, D)$, then choose the more dependency attribute.

Step 5: return the result R

C. Building Classifier

Performance of classifier should be measured by confidence and stability of classification. Because the classification result of higher creditable classifier is more plausible than that of the lower creditable classifier. There is an idea that the weight of voting is decided by the confidence of classification.

Higher creditable classifier should have higher weight of voting [10].

In 1761, Bayes formula is put forward by Thomas Bayes, which has reflected relationship between the prior probability and the posterior probability.

The level of the text classifier confidence can be decided by posterior probability.

$$con_i = p(\omega_j | x_i) \quad (26)$$

Where con_i is represented confidence of classifier H_i ; $p(\omega_j|x_i)$ indicates posterior probability of H_i .

There are T weaker classifiers $H_1, H_2, ...H_t$ and their confidence values are $con_1, con_2 ..., con_t$, so voting weight of each weaker classifier can be defined as follows:

$$weig_i = 1 - \frac{1/con_i}{\sum_{i=1}^{T} 1/con_i} - \frac{T-2}{T} \quad (27)$$

Where $weig_i$ is represented voting weight of H_i ; T is the total number of weaker classifiers.

It is obvious that voting weight of the weaker classifier is higher when the confidence of weaker classifier is higher, and equation (27) meets the condition of normalized.

$$\sum_{i=1}^{T} weig_i = 1 \quad (28)$$

The training set is $S = \{x_1, y_1\}, \{x_2, y_2\}, ...\{x_m, y_n\}$, and classifies label is $y_i = \{1, -1\}$ .The rule of confidence voting can be defined as follows:

$$y = \frac{1}{T} \sum_{i=1}^{T} weig_i \times y_i \quad (29)$$

As pointed out by Breiman et al. [4], stability is a key factor for Bagging to improve accuracy of classifier ensembles. Bagging can improve accuracy of instability machine learning algorithm. Theoretical and empirical results suggest [11] that kNN is an unstable algorithm because it is very sensitive to the additions and deletions of attributes.

Reference [5] proposes an improved Bagging called Attibute Bagging (AB) which is used attribute subsets to produce training sets.

In AB, There are T weaker classifiers are built by kNN and each classifier's training set is produced by randomly drawn attributes with replacement. Many of the original attributes may be repeated in the resulting training set while others may never appear. The new training set is as the same size as the original training set. When a new instance comes, the result is obtained by voting outputs of T classifiers.

In this paper the same weight voting of Attribute Bagging is improved by confidence voting method. The confidence voting weights of weaker classifiers are calculated by equation (27). In this way the confidence Attribute Bagging can be obtained.

The main idea of Confidence Attribute Bagging is that kNN is used to build the weaker classifier. The result of classification is gotten by confidence voting outputs of weaker classifiers.

Confidence Attribute Bagging (CAB) for Chinese architecture document categorization is described as follows:

Algorithm 3 Confidence Attribute Bagging.

Input: training set S, weaker classifier kNN

Where $S = \{x_1, y_1\}, \{x_2, y_2\}, ...\{x_m, y_n\}$, Total m documents are divided into n categories, x_i is an instance, which is denoted by d condition attributes, $A = \{a_1, a_2 ..., a_d\}$. y_i is the text category, namely the decision attribute.

Output: result of classifier h^*

Step 1: decide the J and T (J is the number of attributes drawn from the attribute set; $j < d$ .T is the times that training is carried on).

Step 2: for i=1 to T do

Step 2.1 attribute set A_t can be obtained by drawing j attributes with replaced form A, a new training set S_t is gotten through drawing attribute set A_t of each instance in S

Step 2.2 take S_t as the training set of kNN algorithm, weaker classifier h_t is obtained.

Step 2.3 according to result of classify, the con_t is calculated by equation (26)

Step2.4 the voting weight is gotten by

$$weig_i = 1 - \frac{1/con_i}{\sum_{i=1}^{T} 1/con_i} - \frac{T-2}{T}$$

Step 3: To the unknown document X, the category h^* is gotten by the voting of T results of the weaker classifiers.

$$H(x) = sign(\sum_{i=1}^{T} weig_i h_i(x))$$

Step 4: Return the result h^* .

V. EXPERIMENTAL RESULTS

Classification effectiveness is usually measured in terms of precision, recall and F1 [12], while Recall, Precision and F1 are defined as follows:

$$recall = \frac{a}{a+c} \quad (30)$$

$$pecision = \frac{a}{a+b} \quad (31)$$

$$F_1 = \frac{recall \times precision \times 2}{recall + precision} \quad (32)$$

a- the number of documents correctly assigned to this category.

b- the number of documents incorrectly assigned to this category

c- the number of documents incorrectly rejected from this category.

We selected 4000 web pages about architecture from Internet as Chinese architectural corpora, and there are four categories including architecture, urban planning, civil engineering and water supply . Each category consists of 1000 documents, and we select 700 documents as training dataset to build classification model, and 300 documents as testing dataset. We use the

ICTCLAS (http://www.ictclas.org/) to make Chinese words segmentation.

The mission of the experiment is comparing two algorithms AB and our method with Chinese architectural documents categorization. The DF method is used to selected feature in AB algorithm. The results of the experiment are shown in table IV .table V and table Ⅵ.

TABLE IV. COMPARISON OF PRECISION

Algorithm	Architecture	Urban planning	Civil engineering	Water supply
Our method	**0.912**	**0.889**	**0.907**	**0.921**
AB	0.883	0.872	0.892	0.910

TABLE V. COMPARISON OF RECALL

Algorithm	Architecture	Urban planning	Civil engineering	Water supply
Our method	**0.891**	**0.875**	**0.887**	**0.906**
AB	0.864	0.854	0.861	0.875

TABLE VI. COMPARISON OF F_1

Algorithm	Architecture	Urban planning	Civil engineering	Water supply
Our method	**0.901**	**0.881**	**0.897**	**0.913**
AB	0.873	0.863	0.876	0.892

From the tables above, we can find that recall rate, precision rate and F1 of our method are better than that of Attribute Bagging. one reason is that our algorithm can keep the low frequency Chinese Architectural terminologies which are very important to text classification, another reason is confidence voting can improve classification accuracy in machine learning. So the conclusion can be drawn that our method can improve classification accuracy for Chinese architectural document.

VI. CONCLUSION AND FUTURE WORKS

In this paper, a novel knowledge discovery method based on rough set and Confidence Attribute Bagging is proposed. The feature selection is based on rough sets and the Chinese architecture document classifier is designed by CAB algorithm. Experiments show that our method gets lower errors and better performance than Attribute Bagging and DF method. Our future work is applying this approach to automatic classification on large-scale web pages.

ACKNOWLEDGMENT

The work is supported by the National Natural Science Foundation of China under Grant No. 50878176; also by Youth Science Foundation of Xi'an University of Architecture and Technology Grant No.QN0827.

REFERENCES

[1] Fabrizio S. Machine Learning in Automated Text Categorization . J of the ACM , vol.34,No. 1, pp.1-47, 2002.
[2] Dietterich T G. Machine learning research: four current directions [J]. AI Magazine, 1997, 18(4): 97-136.
[3] SHEN Xue-hua, ZHOU Zhi-hua, WU jian-xin et al. Survey of Boosting and Bagging [J]. Computer Engineering and Application, 2000, 36(12), 31-32.
[4] Beriman L. Bagging Predictors. Machine learning , vol.24,No 2, pp.123-140,1996.
[5] Bryll R.,Gutierrez O.R.,Quek F.. Attribute Bagging: Improving accuracy of classifier ensembles by using random features subsets. Pattern Recognition Letters, 2003, 36(6):1291-1302.
[6] Kjersti Aas and Line EiKvil.Text Categorisation： A Survey.Norway：Norwegian Computing Center，June，1999.
[7] Pawlak, Z. Rough sets: theoretical aspects of reasoning about data. Kluwer Academic Publishers, Dordrecht, 1991
[8] Wang guoyin. Rough sets and knowledge get . Xi'an jiao tong university publishers,2001.(in Chinese)
[9] Miao duoqian, Hu guirong. A Heuristic Algorithm for Reduction of Knowledge . Journal of computer research & development ,vol.36,No.6, pp.681-683.June l999 (in Chinese).
[10] Langley P, Iba W. Average-case Analysis of Nearest Neighbor Algorithm [A]. Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence[C], Morgan Kaufmann Publishers San Francisco USA 1993:889-894.
[11] YAN Ji-Kun, ZHENG Hui, WANG Yan et al. voting by confidence [J]. Chinese Journal of Computer. 2005, 28(8), 1308-1312. in Chinese).
[12] Songbo Tan，Yuefen Wang ,Xueqi Cheng. An Efficient Feature Ranking Measure for Text, *SAC'08*,pp.407-413, Fortaleza, Ceará, Brazil March 16-20, 2008,.

Xiang Zhang, Xi'an Shaanxi, China, born in 1972, received the master degree in computer application technology from Northwestern University in 2003.

He is now an associate professor and master student supervisor in School of Information and Control Engineering of Xi'an University of Architecture and Technology.

His research areas are data mining and intelligent information processing.

Changhua Li was born in Yinchuan, China in 1963. He received his Ph.D. in information and communication engineering from Xidian University 2002.

He is now the professor and PH.D. Supervisor of school of information & automation engineering in Xi'an university of architecture & technology.

His research interests are image processing, computer graphics, data mining and digital architecture.

Lili Dong, Fuzhou Fujian, China, born in 1960, graduated from Northwestern University in 1983.

She is now the professor and master student supervisor in School of Information and Control Engineering of Xi'an University of Architecture and Technology. She is a member of Shaanxi Computers society.

Her research areas are distributed system and data mining.

Na Ye, Xi'an Shaanxi, China, born on April 28th, 1979, received the master degree in computer application technology from Xi'an University of Architecture and Technology in 2005.

She is now a lecturer in School of Information and Control Engineering of Xi'an University of Architecture and Technology.

Her research areas are SOA and data mining.