



Improving the Accuracy and Efficiency of CBA Algorithm



Zheng Tan¹, Hanhu Wang¹, Mei Chen¹, Xiaoping Zhang²

¹Computer Science & Technology Department

Guizhou University

Guiyang, China

tanzheng6907@163.com

²Guizhou Science and Technology Information Institute

Guiyang, China

xpzhang@gzgwyy.gov.cn

ABSTRACT: Classification is an important research topic in data mining field, and it is one of main task of data mining. CBA (Classification Based on Associations) is a classification algorithm integration association rule mining and classification. CBA has been widely used in data mining areas because it has high classification accuracy and strong flexibility at handling unstructured data. However, when the samples become more and more large and characteristic attributes become more and more numerous, CBA algorithm becomes much lower. In this paper, an improved CBA algorithm based on rough set is proposed to improve both accuracy and efficiency. The improved CBA algorithm applies rough set theory to reduce attributes, and prune candidate rules with PEP method. Experimental result illustrate that the improved CBA algorithm is more efficient than CBA, and it has higher accuracy than CBA and C4.5.

Keywords: Data mining, CBA classification, Rough set, Attributes induction, PEP

Received: Received 02 December 2009, Revised 03 February 2010, Accepted 9 February 2010

© 2009 D-Line. All rights reserved.

1. Introduction

Building accurate and efficient classifiers for large databases is one of the essential tasks of data mining. Given a set of cases with class labels as a training set, classification is to build a classifier to predict future data objects for which the class label is unknown. At present, classification is widely applied in medical diagnosis, financial cheat analysis and text classification. Previous studies have developed many techniques for building classifiers, such as decision trees, ANN, CBA(Classification Based on Association) [1] and Bayes. CBA is widely used in many fields because of its simplicity and high classification accuracy.

In recent years, extensive research has been carried to improve the CBA, e.g., ADT, CMAR [2] and CPAR[3]. These algorithms improved the classification accuracy and have better classification efficiency [4] than CBA and C4.5 [5]. However, these algorithms neglect the dependencies between the characteristic attributes. When the number of samples is huge and the characteristic attributes is numerous, these algorithm will generate a huge of redundant rules which affect the classification efficiency.

Rough set [6] was originally proposed by Pawlak as a mathematical approach to handle imprecision, vagueness and uncertainty in data analysis. Now, Rough set has been widely applied in data mining [7, 8, 9]. An important application of rough sets theory is attribute reduction. Based on Rough sets theory, the redundant attributes can be successfully eliminated on condition that the reduced decision table has the equal classification ability with the original decision table. This is useful for us to make correct and compact decisions.

In this paper, we propose an improved CBA algorithm for accurate and efficient classification and make the following contributions: First, we apply rough sets to reduce characteristic attributes in decision table, and then the redundant attributes will be deleted. It can avoid the generation of a large number of redundant rules. So we can generate correct and compact rules. Second, in the procedure of rule generation, we apply PEP (pessimistic errors pruning) to prune biased rules which affect the accuracy of classification.

The rest of the paper is organized as follows. Section 2 devotes to the attribute reduction based on rough set theory. Section 3 discusses how to prune candidate rules with PEP method. Section 4 presents the improved CBA algorithm. The experimental results are reported in Section 5. The conclusions are drawn in Section 6.

2. Attribute reduction based on rough set theory

The theory of rough sets has emerged as a major mathematical tool for managing uncertainty that arises from granularity in the domain of discourse, i.e., from the indiscernibility between objects in a set, and has proved to be useful in a variety of KDD processes [10]. A fundamental principle of a rough set-learning system is to discover redundancies and dependencies between the given features of a problem to be classified [11]. It approximates a given concept from below and from above, using lower and upper approximations.

Attribute reduction is very important for data mining. With attribute reduction, the redundant knowledge can be successfully eliminated from the databases, and the hidden laws as well as relationships among the attributes can be discovered.

In this section, we develop an attribute reduction algorithm base on rough set.

2.1 Basic concepts of rough set theory

In rough set theory, an information system S is denoted by $S = \langle U, A, V, f \rangle$, where $U = \{x_1, x_2, \dots, x_n\}$ is a finite set of objects, called universe of discourse; A is a nonempty set of attributes; $V = \bigcup_{a \in A} V_a$, V_a represents the domain of a ; and $f: U \times A \rightarrow V$, called an information function, assigns an attribute value to each x in U , i.e., $f(x, a) \in V_a$ for all $x \in U$, $a \in A$.

Definition 1 Let $S = \langle U, A \rangle$ be an information system, for each subset $B \subseteq A$, definite a discernibility relation $IND(B)$:

$IND(B) = \{(x, y) \in U \times U \mid \forall a \in B, f(x, a) = f(y, a)\}$ for each object $x \in U$, the equivalence class of x is defined as

$$[x]_B = \{y \mid \forall y \in U, (x, y) \in IND(B)\}.$$

Definition 2 In an information system, $S = \langle U, A \rangle$, for each subset $X \subseteq U$ and an attribute set $R \subseteq A$, the lower and upper approximations of X are respectively defined as follows:

$$\underline{R}X = \{Y \in U / R \mid Y \subseteq X\};$$

$$\overline{R}X = \{Y \in U / R \mid Y \cap X \neq \emptyset\};$$

Definition 3 Let $S = \langle U, C \cup D \rangle$ is a decision table, C is the condition attribute set, D is the decision attribute set, for an attribute set $B \subseteq C$, the positive region of D relative to B is defined as:

$$POS_B(D) = \{\underline{B}X \mid X \in U / IND(D)\}.$$

$POS_B(D)$ actually is the object set which can be accurately divided into equivalence class of D according to the information of U/B .

If $B \subseteq C$, and $POS_B(D) = POS_C(D)$, then B is called the relative reduction of C .

Definition 4 Let $S = \langle U, C \cup D \rangle$ is a decision table with $U = \{x_1, x_2, \dots, x_n\}$. The discernibility matrix of S is an $n \times n$ matrix, denoted by $M(S)$ and defined as

$$M_{ij} = \{a \in C \mid f(x_i, a) \neq f(x_j, a) \text{ and } w(x_i, x_j)\}$$

Given two objects x_i, x_j , M_{ij} is the attribute set which can discern x_i and x_j . It is obvious that $M(S)$ is a symmetrical matrix. Therefore, $M(S)$ can be simplified to its upper triangular form. If there exist an element concluding only one attribute in $m(S)$, then the element is the unique attribute which can discern the two relative objects. We call it core attribute. Our main work is to reduce the none-core attributes.

Definition 5 Let $S = \langle U, C \cup D \rangle$ is a decision table, the discernibility function of S is defined as:

$$\Delta = \prod_{(x_i, x_j) \in U \times U} \sum M_{ij}$$

It can be known that the discernibility function Δ has following character: After Δ is converted to minimum disjunctive normal form, all the conjunctive forms of Δ are all the reductions of original decision table [12].

2.2 The attribute reduction algorithm based on roughset

The main idea of the algorithm is: Firstly, compute the discernibility matrix of decision table S according to the samples; Secondly, obtain all core attributes from $M(S)$, and set the element to \emptyset which conclude core attribute; For the other elements in $M(S)$, construct the relative Boolean form P_{ij} ; Thirdly, absorption and distribution laws are employed to convert the Boolean expression from conjunctive form to disjunctive form. At last, the discernibility function Δ are gained and converted to disjunctive normal form, of which all the conjunctive forms are all the reductions of the decision table S . The attribute reduction algorithm (called algorithm 1) is illustrated in Figure 1.

```

Input: the decision table  $S = \langle U, C \cup D \rangle$ 
Output: reduced attribute set  $C'$ 
(1)// Compute the discernibility matrix  $M(S)$ ,
    extract core attributes CORE
    For  $i=2$  to  $n$  do
        For  $j=1$  to  $i-1$  do
             $M_{ij} = \{ a \in C \mid f(x_i, a) \neq f(x_j, a) \text{ and } w(x_i, x_j) \}$ ;
            If  $|M_{ij}| = 1$ 
                Then  $CORE = CORE \cup M_{ij}$ ;
            End for
        End for
    End for
(2)// Set the elements concluding core attributes
    to  $\emptyset$ , construct relative conjunctive form
    of the other elements.
    For  $i=2$  to  $n$  do
        For  $j=1$  to  $i-1$  do
            If  $M_{ij} \cap CORE \neq \emptyset$  Then  $M_{ij} = \emptyset$ ;
            Else  $P_{ij} = \bigwedge_{a_k \in M_{ij}} a_k$ ;  $P = P \vee P_{ij}$ ;
        End for
    End for
(3)// Convert  $P$  to disjunctive form  $P'$ .
     $P' = CNF(P)$ ;
(4)// Compute the discernibility function  $\Delta$ ,
    and convert  $\Delta$  to disjunctive normal form.
     $\Delta = P' \wedge CORE$ ;
     $Q = DNF(\Delta)$ ;

```

Figure 1. The attribute reduction algorithm

2.3 An example for attribute reduction

Consider Table 1 as a decision information table, where $\{a, b, c\}$ is the condition attribute set, $\{d\}$ is the decision attribute set. There are 6 samples in decision table.

u	a	b	c	d
1	2	2	0	1
2	1	2	0	0
3	1	2	0	1
4	0	0	0	0
5	1	0	1	0
6	2	0	1	1

Table 1. Decision table

Firstly, according to Definition 4, we can compute the discernibility matrix (M_{ij}) of the decision table as follows:

$$\begin{bmatrix} a \\ a \\ a,b & a,b & a,b \\ a,b,c & b,c & b,c \\ & a,b,c & a,b,c & a,c & a \end{bmatrix}$$

From M_{ij} , we can get the core attributes CORE, CORE = {a}. There is only one attribute set {b,c} not concluding the core attributes. Then construct the disjunctive form P, $P = b \vee c$. Because P has only one sub disjunctive form, it is no need to convert P to disjunctive normal form. At last, we can get the discernibility function Δ , $\Delta = (b \vee c) \wedge a$. The disjunctive normal form of Δ is $\Delta = (b \wedge a) \vee (c \wedge a)$. Thus, the condition attributes of the table x can be reduced to {a,b} or {a,c}. The reduced decision table is showed as follows:

u	a	b	D
1	2	2	1
2	1	2	0
3	1	2	1
4	0	0	0
5	1	0	0
6	2	0	1

Table 2. Reduced decision table T1

u	a	b	C
1	2	2	0
2	1	2	0
3	1	2	0
4	0	0	0
5	1	0	1
6	2	0	1

Table 3. Reduced decision table T2

The reduced decision table T1 and T2 has the equal classification ability with the original decision table.

3. Rule pruning

The number of rules generated by CBA_RG algorithm can be huge to make the classification effective and efficient; we need to prune rules to delete redundant and noisy information [13]. There are various rule pruning methods designed to reduce

the classifier's size and to increase the accuracy of classification [14]. We adapt PEP method to prune rules which affect the accuracy of classification. PEP method was proposed by Quinlan [15] that aims to avoid the necessity of a separate test data set. It is based on the number of errors and the size of the training sample.

A candidate rule generated by CBA_RG is of the form: $r = \langle \text{condset}, y \rangle$, where $\text{condset} = \{(C_1, a_1), (C_2, a_2), \dots, (C_n, a_n)\}$ is a finite set of items, called condition set, $y \in V_d$, V_d represents the domain of decision attribute d . If there exist a rule $r = \langle \text{condset}, y \rangle$, which satisfies $|r.\text{condset} \cdot r.\text{condset}| = 1$, the r is called the father rule of r .

The errors number of the rule r is

$$e(r) = r.\text{condsupCount} - r.\text{rulesupCount}$$

An estimate of the miss-classification number is

$$e' = |C| (e(r) + 1/2).$$

Where $C = r.\text{condse} - \bar{r}.\text{condset}$.

The standard error is calculated in this wy:

$$SE(e'(r)) = \sqrt{\frac{e'(r) * (\text{cond sup Count} - e'(r))}{\text{cond sup Count}}}$$

Accordingly, for the rule \bar{r} , the estimate of the miss-classification number

$$e'(\bar{r}) = e(\bar{r}) + 1/2.$$

PEP suggests pruning the rule r if its correct number of miss-classification is greater than that for \bar{r} , e.g. $e'(\bar{r}) < e'(r) + SE(e'(r))$. For example, we have two rule r and \bar{r} (the father rule of r) as follows:

$$r: \langle \{(A,1), (B,1), (C,1)\}, (\text{class}, 1) \rangle, r.\text{condsupCount} = 80, r.\text{rulesupCount} = 70$$

$$\bar{r}: \langle \{(A,1), (B,1)\}, (\text{class}, 1) \rangle, \bar{r}.\text{condsupCount} = 100, \bar{r}.\text{rulesupCount} = 80$$

Assume the attribute C has two values, e.g. $|C| = 2$. Then we can get:

$$e'(\bar{r}) = 20 + 1/2 = 20.5.$$

$$e'(r) = 2 * 10 + 2 / |C| = 21.$$

$$SE(e'(r)) = (21 * (100 - 21) / 100)^{1/2} \approx 4.$$

Since $21 + 4 = 25$, which is greater than 20.5, the rule r will be pruned.

We adapt PEP method because it has following advantages: the same training set is used for both generating and pruning the CAR (Classification Association Rule), and it is much quick because it only has to make one pass and looks at each rule only once. Our experimental results show that this pruning method is effective and efficient.

4. The improved CBA algorithm

4.1 CBA algorithm

CBA is a Classification Association Rule Mining (CARM) algorithm. It consist of two parts, a rule generator (called CBA_RG), which is based on algorithm Apriori for finding association rules, and a classifier builder (called CBA_CB).

It builds a classifier as follows:

1. Rule Generating (CBA_RG)

Based on the framework of Apriori, find all class association rules (CARs) in the form of $R: P \rightarrow c$, where P is a pattern in the training data set, and c is a class label, such that $\text{sup}(R)$ and $\text{conf}(R)$ pass the give support and confidence thresholds respectively.

2. Classifier Building (CBA_CB)

Firstly, sort the set of generated CARs according to the precedence. This is to ensure that the highest precedence rules will be chosen for the classifier. Secondly, select rules for the classifier from CARs following the sorted sequence. For each rule r , remove those cases covered by rule r ; if there is no one case covered by rule r , r will not be chosen for the classifier. Thus each

rule of CARs covers at least one case. The majority class in the remaining data is chosen as the default class. When there is no rule or no training case left, the rule selection process is completed. Finally, discard those rules in the set of CARs that do not improve the accuracy of the classifier. The undiscarded rules and the default class of the last rule form the classifier.

The CBA algorithm can be quite effective when the attributes of the data are equally important. But it can be less effective when many of the attributes are misleading or irrelevant to the classification.

4.2 The improved CBA algorithm

To improve the accuracy and efficiency of CBA, we proposed an improved CBA algorithm based on rough sets.

Our proposed algorithm build the classifier as follows: Firstly, the redundant attributes of the training set are reduced by algorithm 1, this is useful to make correct and compact decisions. Secondly, generate CARs with Apriori algorithm, and the CARs that affect the efficient of classification will be pruned with PEP method. Finally, the classifier will be built referring to CBA_CB. The improved CBA algorithm (called algorithm 2) is illustrated in Figure 2.

```

Input: the decision table  $S = \langle U, C \cup D \rangle$ 
Output a Classifier
(1)// Reduce attributes with algorithm1.
   C'=Reduction(S);
   Del(S);
(2)// Generate CARs.
   F1={large 1_ruleitem};
   CAR1=genRule(F1);
   For(k=2;Fk-1≠∅;k++)do
       Ck=candidateGen(Fk-1);
       Count_rulesup(Ck);
       Fk={c ∈ Ck | c.rulesupCount ≥ minsup};
       prCARk=PEP(CARk);
   End for
   prCARs=  $\bigcup_{i=1 \text{ to } k}$  prCARi;
(3)// Build the classifier referring to CBA_CB.
   Classifier=CBA_CB(prCARs);

```

Figure 2. The improved CBA algorithm

5. Experimental results

To evaluate the accuracy and efficiency of the improved CBA, we have performed an extensive performance study. In this section, we report our experimental results on comparing the improved CBA against CBA and C4.5. It shows that the improved CBA outperforms both CBA and C4.5 in terms of average accuracy and efficiency.

All the experiments are performed on a 2.8 Hz Pentium-4 PC with 1GB main memory, running Microsoft Windows/NT. CBA and C4.5 were implemented by their authors, respectively. In the experiments, the parameters of the three methods are set as follows.

For C4.5, all parameters are default values. We test both C4.5 decision tree method and rule method. Since the rule method has better accuracy, we only report the accuracy for rule method.

For CBA, we set support threshold to 1% and confidence threshold to 50% and disable the limit on number of rules. Other parameters remain default.

For the improved CBA, The support and confidence thresholds are set as same as CBA.

We test 29 data sets from UCI ML Repository. For CBA and the improved CBA, data set were cleaned and discretized before using. For C4.5, which is capable of using continuous values, both discretized and nondiscretized data sets were used and the best results were presented.

Data set	#attr	#rec	C4.5	CBA	Improved CBA
Anneal	38	898	94.8	97.9	98.1
Austral	14	690	84.7	84.9	85.7
Auto	25	205	80.1	78.3	79.5
Breast	10	699	95	97	97.7
Cleve	13	303	78.2	82.8	82.3
Crx	15	690	84.9	84.7	84.9
Diabetes	8	768	74.2	74.5	75.6
German	20	1000	72.3	73.4	75.2
Glass	9	214	68.7	73.9	74.4
Heart	13	270	81.2	82.7	84.3
Hepatic	19	155	80.6	81.8	81.9
Horse	22	368	83	84.4	85.2
Hypo	25	3163	99.2	93.9	99.1
Liver	22	73n	74.8	76.2	77.3
Lono	34	351	90	92.3	92.7
Iris	4	150	95.3	94.7	95.1
Labor	16	57	79.3	86.3	87.5
Led7	7	3200	73.5	71.9	72.7
Lung	32	853	81.4	83.5	84.8
Lymph	18	418	73.5	77.8	80.6
Pima	8	768	75.5	72.9	75.1
Sick	29	2800	98.5	97	98.3
Sonar	60	208	70.2	77.5	78.9
Tic-tac	9	958	99.4	99.6	99.5
Vehicle	18	846	72.6	68.7	69.8
Waveform	21	5000	78.1	80	82.9
Wine	13	178	92.7	95	95
Zoo	16	101	92.2	96.8	97.8
Average			82.9	84.5	85.4

Table 1. Accuracy results for each algorithm

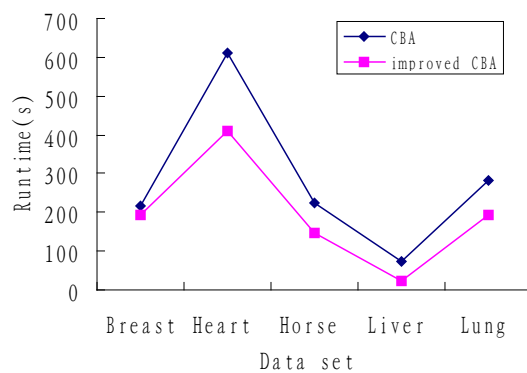


Figure 3. The comparison of CBA and improved CBA on runtime

As can be seen from the Table 1, the improved CBA outperforms both C4.5 and CBA on accuracy. Furthermore, when the number of attributes and samples of the training set are huge, the advantage of the improved CBA is more obvious, e.g. Breast and Lung, the improved CBA wins C4.5 over 3% in accuracy.

To test the efficiency of the improved CBA, we compare the runtime of CBA and the improved CBA on five data sets. The results are shown in Figure 3. As can be seen from the table, the run time of the improved CBA is about half of the CBA's. It indicates that the improved CBA is much more efficient than CBA.

6. Conclusions

In order to improve the efficiency of CBA, a new CBA algorithm based on rough set theory is proposed in the paper. This method applies rough set to reduce redundant attributes in database, and adapt PEP method to prune rules which affect the accuracy of classification. Experimental results proved that the improved CBA is more efficient than CBA and it has higher accuracy compared with CBA and C4.5. Therefore, the classifier build by this method is effective and efficient.

Through analysis of the experiment result, we found that the improved algorithm performs well when handling with data set which has little continual attributes, but when data set conclude many continual attributes, the algorithm's performance is not ideal. In our future work, we will focus on how to discretizate the continual attribute effectual according respective data set.

7. Acknowledgment

This research is supported by Industry Plans Projects of Guizhou Province in China and Informatization Special Fund Project of Guizhou Province in China.

References

- [1] Liu, B. Hsu, W., Ma, Y. (1998). Integrating classification and association rule mining, In: KDD'98, New York, p.80-86
- [2] Li, W., Han, J., Pei, J. (2001). CMAR: Accurate and efficient classification based on multiple class-association rules, In: Proc. of the ICDM, p. 369-376.
- [3] Yin, X., Han, J. (2003). CPAR: Classification based on predictive association rules. In: 2003 SIAM International Conference on Data Mining (SDM'03). San Fransisco, CA. p. 331-335
- [4] Janssens, D.,etal. (2003). Adapting the CBA algorithm by means of intensity of implication. In: the First International Conference on Fuzzy Information Processing Theories and Application. Beijing, China: p. 397-403
- [5] Quinlan, J. R (1993). C4.5: Programs for Machine Learning, Morgan Kaufmann.
- [6] Pawlak, Z. (1991). Rough Sets-Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston.
- [7] Pawlak, Z. (2008). Rough sets and intelligent data analysis", Information Sciences, vol. 147. 2002, p. 1-12
- [8] Huawang, SHI (2008). The Risk Early-warning of Hidden Danger in Coal Mine Based on Rough Set-neural network, In: Proceeding of the 2nd International Conference on Risk Management and Engineering Management, 2008, p. 314-317
- [9] Shi, Huawang Li, Wanqing Meng, Wenqing (2008). A New Approach to Construction Project Risk Assessment Based on Rough Set and Information Entropy, 2008 International Conference on Information Management, Innovation Management and Industrial Engineering, p. 187-190
- [10] De Cock, Martine., Cornelis, Chris., Kerre, Etienne E. (2007). Fuzzy Rough Sets: The Forgotten Step, IEEE Transactions on Fuzzy Systems. 15 (1) 121-130
- [11] Zhong, N., Dong, J.Z (2001). Using rough sets with heuristics for feature selection, Journal of Intelligent Information Systems, 16, p. 199-214
- [12] Lashin, E.F., Medhat, T. (2005). Topological reduction of information systems, Chaos, Solitons and Fractals, v. 25. p. 277-286
- [13] Breslow, L A, Aha, D. W (1997). Simplifying decision trees: a survey[J]. Knowledge Engineering Review, 1997, 12 (1) 1-40.
- [14] Coenen, Leng (2004). An evaluation of approaches to classification rule selection, In: Proc. of the IEEE ICDM. p.359-362
- [15] Quinlan, J R (1987). Simplifying decision trees, International Journal of Man Machine Studies, 27(3) 221-234