International Journal of Database Theory and Application
Vol.9, No.10 (2016), pp.199-214
http://dx.doi.org/10.14257/ijdta.2016.9.10.17

Exploring the Boundary Region for Attribute Reduction in Inconsistent Decision Tables

Yuhui Chang and Jin Qian

School of Computer Engineering, Jiangsu University of Technology, Changzhou, Jiangsu, China
cyh@jsut.edu.cn, qjqjlqyf@163.com

Abstract

Attribute reduction is one of the key issues for data preprocess in data mining. Many heuristic attribute reduction algorithms based on discernibility matrix have been proposed for inconsistent decision tables. However, these methods are usually computationally time-consuming. To address this issue, the derived consistent decision tables are defined for different definitions of relative reducts. The computations for different reducts of the original inconsistent decision tables are converted into the computations for their corresponding reducts of the derived consistent datasets. The relationships among different core sets and attribute reducts are further discussed. The relative discernibility object pair and the more optimal relative discernibility degree from view of the boundary region are designed to accelerate the attribute reduction process. An efficient attribute reduction framework using relative discernibility degree is proposed for large datasets. Experimental results show that our attribute reduction algorithms are effective and feasible for large inconsistent datasets.

***Keywords**: Rough set, attribute reduction, discernibility matrix, boundary region, counting sort*

1. Introduction

In recent years, many datasets increase dramatically in the number of objects and in the condition attributes. In general, there exist some irrelevant and redundant attributes. It is found that such attributes will decrease the generalization power of the learned classifiers for some data mining and machine learning tasks. Attribute reduction, also called feature selection, is often carried out as a preprocessing step to find a minimum subset of attributes that provides the same descriptive or classification ability as the entire set of attributes. In recent years, attribute reduction with rough set theory [14] has attracted many attentions. There are many types of attribute reducts in inconsistent decision tables [7,25,8,11]. Pawlak proposed the classical attribute reduction, which preserved the positive region and extracted the deterministic decision rules. However, since there has noise data, most decision tables are inconsistent. The Pawlak reduct cannot preserve non-deterministic decision information for inconsistent decision tables. To this end, Kryszkiewicz [7] proposed assignment reduction and distribution reduction which can preserve the non-deterministic information in the boundary region and mined some possible decision rules. In an inconsistent decision table, assignment reduction can maintain unchanged the possible decisions for an arbitrary object, while distribution reduction can preserve the class membership distribution for all of the objects. Zhang *et al.* [25] had proposed the maximum distribution reduction which can maintain unchanged the maximum decision classes for all of the objects. Thus, we can derive the maximum decision rules from an inconsistent decision table. Li *et al.* [8] proposed an efficient heuristic attribute reduction algorithm for the Pawlak reduct of three types of derived

ISSN: 2005-4270 IJDTA

consistent decision tables. Miao *et al.* [11] systematically studied the relationships among different relative reducts using discernibility matrix.

Different efficient attribute reduction algorithms for different models of rough sets and different types of rough sets have been developed to acquire an optimal reduct or the set of all reducts. As we all know, finding all reducts is NP-Hard problem. Therefore, we usually acquire a reduct. In heuristic search strategy among attribute reduction methods [27], we first start with an empty set, and then add the most optimal attribute into the reduct one by one. Most existing heuristic attribute reduction algorithms employed dependency function [15,21,9], information gain [10,15], and other measures [13] to select the most significant attribute in each round. For Pawalk reduct, Qian and Liang [15] proposed positive approximation to characterize the granulation structure of a rough set using a granulation order for improving the time efficiency of the positive region reduction algorithm with $O(|C||U|+\sum_{1}^{|C|}|U_i|(|C|-i+1))$. Qian and Miao [16] proposed a fast attribute reduction algorithm based on positive region, discernibility matrix and information entropy with max($O(|C||U|)$, $O(|C|^2|U/C|)$) using counting sort technique. Many researchers began to study attribute reduction algorithms for different relative reducts in inconsistent decision tables. Jiang *et al.* [5] proposed a quick distribution reduction algorithm with $O(|C||U|)$ using Hash technology based on information entropy criteria. Li *et al.* [9] derived three novel types of attribute significance measures and presented quick attribute reduction algorithms for the assignment reduct, the distribution reduct, and the maximum distribution reduct in the inconsistent decision table with $O(|C|^2|U|)$. Another important attribute reduction method for different types of relative reducts utilized the discernibility information from discernibility matrix [18,26] to acquire different relative reducts in inconsistent decision tables. Zhang *et al.* [25] proposed an approach to maximum distribution reduction based on discernibility matrix. Miao *et al.* [11] systematically studied relative reduct construction based on discernibility matrix and presented the generalized discernibility matrix and discernibility function. However, reduct construction methods based on discernibility matrix have high cost of storage with space complexity $O(|C||U|^2)$ for a large decision table with $|U|$ objects and $|C|$ conditional attributes. Thus, storing and deleting the element cells in a discernibility matrix is a time-consuming process. Although discernibility matrix-based methods can find all the reducts, they become not feasible for large datasets. Therefore, an efficient heuristic attribute reduction algorithm is desirable.

Since an efficient heuristic attribute reduction method reflects in three aspects—computing equivalence classes, reducing storage and minimizing the search space, our focus is on how to improve the time efficiency of a heuristic attribute reduction algorithm. In order to accelerate the attribute reduction process, we first emphasize and unify various types of relative reducts in inconsistent decision tables and discuss the relationships among core sets and these relative reducts. Then, we propose a more reasonable uncertainty measure from view of the boundary region—relative indiscernibility degree and employ the counting sort algorithm with time complexity $O(|C||U|)$ to compute the equivalence classes and core attributes. Finally, we propose an efficient attribute reduction framework for inconsistent decision tables with time complexity max ($O(|C|^2|U/C|)$, $O(|C||U|)$). Numerical experiments show that our methods are more efficient and feasible for large inconsistent decision tables.

The rest of this paper is organized as follows. In Section 2, we review necessary concepts in rough set theory. Section 3 defines the derived consistent decision table and discusses the relationships among core sets and five relative reducts. Section 4 designs a more optimal relative discernibility degree from view of the boundary region and presents an efficient attribute reduction framework for inconsistent decision tables. Section 5 gives some numerical experimental results to validate the efficiency and effectiveness of the proposed algorithms. Finally, we conclude this paper and several issues for future works.

2. Basic Notions

In this section, we review the basic notions of the Pawlak rough set model regarding classification and approximation [11, 23, 25]. For classification tasks, we consider a special information table with a set of decision attributes. Such an information table is called a decision table.

Definition 1. [11] A decision table is defined as: S = (U, $At = C \cup D$, $\{V_a \mid a \in At\}$, $\{I_a \mid a \in At\}$), where $U = \{x_1, x_2, \ldots, x_n\}$ is a finite non-empty set of objects, At is a finite nonempty set of attributes, $C = \{c_1, c_2, \ldots, c_m\}$ is a set of condition attributes describing the objects, and D is a set of decision attributes that indicates the classes of objects. V_a is a nonempty set of values of $a \in At$, and I_a : $U \to V_a$ is an information function that maps an object in U to exactly one value in V_a. $a(x)$ denotes the value of attribute a for object x.

Let $A \subseteq U \times U$ an equivalence relation on U. The equivalence relation A determines a partition of U, denoted by $U / IND(A)$, or simply π_A. The equivalence class of $U / IND(A)$ containing object x is given by $[x]_{IND(A)} = \{y \in U \mid (x, y) \in IND(A)\}$. For simplicity, we write $[x]_A$ instead of $[x]_{IND(A)}$ if $IND(A)$ is understood. Assume that D = $\{d\}$ in this paper, where d is a decision attribute which describes the decision for each object, and V_d ={1, 2, …, k }. A table with multiple decision attributes can be easily transformed into a table with a single decision attribute by considering the Cartesian product of the original decision attributes.

Consider a partition $\pi_D = \{D_1, D_2, \ldots, D_k\}$ of the universe U with respect to the decision attribute D and another partition $\pi_A = \{A_1, A_2, \ldots, A_r\}$ defined by a set of condition attributes A. The equivalence classes induced by the partition (*i.e.* , U/IND(A)) are the basic blocks to construct the Pawlak rough set approximations.

Definition 2. For a decision class $D_i \in \pi_D$, the lower and upper approximations of D_i with respect to a partition π_A are defined by Pawlak [30]:

$$\underline{apr}_A(D_i) = \{x \in U \mid [x]_A \subseteq D_i\};$$
$$\overline{apr}_A(D_i) = \{x \in U \mid [x]_A \cap D_i \neq \varnothing\}. \quad (1)$$

For the partition π_D, we can compute its lower and upper approximations in terms of k two-class problems. $POS_A(D)$ indicates the union of all the equivalence classes defined by π_A that each for sure can induce a certain decision. $BND_A(D)$ indicates that the union of all the equivalence classes defined by π_A that each can induce a partial decision.

Definition 3. For a decision table S, a positive region and boundary region of a partition π_D with respect to a partition π_A are defined as:

$$POS_A(D) = \bigcup_{1\le i\le k} \underline{apr}_A(D_i);$$
$$BND_A(D) = \bigcup_{1\le i\le k} (\overline{apr}_A(D_i) - \underline{apr}_A(D_i)). \tag{2}$$

Definition 4. For a decision table S, all objects in $POS_C(D)$ are called consistent objects. The objects in U - $POS_C(D)$, are called inconsistent objects. If $POS_C(D)$ = U, then the decision table is consistent, D_{k+1} = $\varnothing$; other S is inconsistent.

In what follows, we briefly review the concepts of the assignment reduct, the distribution reducts and the maximum distribution reduct [2, 7, 8, 9].

For any $x \in U$, $A \subseteq C$, a decision vector $\mu_A(x)$ is defined as follows:

$$\mu_A(x) = (p(D_1 | [x]_A), p(D_2 | [x]_A), \ldots, p(D_k | [x]_A)) \tag{3}$$

where $p(D_j | [x]_A)$ = $\frac{|D_j \cap [x]_A|}{|[x]_A|}$, j=1, 2, $\cdots$, k.

It is obvious that $\mu_A(x)$ is a probability distribution. On this basis, the maximum decision function $\gamma_A(x)$ is defined as follows:

$$\gamma_A(x) = \{D_j \mid p(D_j | [x]_A) = argmax_{1\le i\le k} p(D_i | [x]_A)\} \tag{4}$$

Obviously, $\gamma_A(x)$ is the decision equivalence classes that has the maximum rough membership of object x. For $\forall x \in U$, the generalized decision function $\delta_A(x)$ is defined as follows:

$$\delta_A(x) = \{D_j \mid p(D_j | [x]_A) > 0\} \tag{5}$$

It can be seen that $\delta_A(x)$ is the family of decision equivalence classes that have an intersection with $[x]_A$.

Definition 5. [11] For a decision table S, the relative indiscernibility and discernibility relations induced by $A \subseteq C$ with respect to D are defined as:

$$IND(A|D) = \{(x, y) \mid \forall a \in A[I_a(x) = I_a(y)] \vee I_d(x) = I_d(y)\};$$
$$DIS(A|D) = \{(x, y) \mid \exists a \in A[I_a(x) \ne I_a(y)] \wedge I_d(x) \ne I_d(y)\}. \tag{6}$$

Definition 6. For an inconsistent decision table S, $A \subseteq C$. Then we have:

(1) A is a Pawlak-consistent set of C if $POS_A(D)$ = $POS_C(D)$. If A is a Pawlak-consistent set and no proper subset of A is a Pawlak-consistent set, then A is referred to as a Pawlak reduct of S.

(2) A is an assignment-consistent set of C if $\forall x \in U$, $\delta_A(x)$ = $\delta_C(x)$. If A is an assignment-consistent set and no proper subset of A is an assignment-consistent set, then A is referred to as an assignment reduct of S.

(3) A is a maximum distribution-consistent set of C if $\forall x \in U$, $\gamma_A(x)$ = $\gamma_C(x)$. If A is a maximum distribution-consistent set and no proper subset of A is a maximum distribution-consistent set, then A is referred to as a maximum distribution reduct of S.

(4) A is a distribution-consistent set of C if $\forall x \in U$, $\mu_A(x)$ = $\mu_C(x)$. If A is a distribution-consistent set and no proper subset of A is a distribution-consistent set, then A is referred to as a distribution reduct of S.

(5) A is a relative indiscernibility relation-consistent set of C if $IND(A|D)$ = $IND(C|D)$. If A is a relative indiscernibility relation-consistent set and no proper subset of A is a relative indiscernibility relation-consistent set, then A is referred to as a relative indiscernibility relation reduct of S.

3. The Relationships among Attribute Reduction Algorithms

According to Definition 6, we can design many attribute reduction algorithms to derive different reducts. In order to emphasize and unify various types of relative reducts in inconsistent decision tables, we first define the derived consistent decision table.

Definition 7. For an inconsistent decision table S, $A \subseteq C$, the derived consistent decision table $S_{\square}$ is defined as

$$S_{\square} = (U, C \cup \{d_{\square}\}, \{V_a \mid a \in C \cup \{d_{\square}\}\}, \{I_a \mid a \in C \cup \{d_{\square}\}\}) \tag{7}$$

where $d_{\square}$ ($\square$ = {p, μ, γ, δ, ω }) denotes the decision attribute for the Pawlak reduct, distribution reduct, maximum distribution reduct, assignment reduct and relative indiscernibility relation reduct, respectively.

In order to efficiently deal with inconsistent decision tables, we convert the computation of five types of relative reducts for the original inconsistent decision table into the computation for their corresponding reducts for the derived consistent decision tables. Thus, any an efficient heuristic attribute reduction algorithm for Pawlak reduct can be used to reduce computational costs. In what follows, we discuss the relationships among different core sets and different definitions of relative reducts.

3.1. The Relationships among Five Core Sets

In order to improve the performance of an attribute reduction algorithm, many researchers [16, 22, 28] study core attributes to reduce search space. Here we define the core set for the derived consistent decision table.

Definition 8. For a derived consistent decision table $S_{\square}$, $A \subseteq C$, the discernibility matrix $M_{\square}$ = $(m_{ij}^{\square})$ is defined as

$$m_{ij}^{\square} = \{a \mid a \in C, I_a(x_i) \neq I_a(x_j) \wedge d_{\square}(x_i) \neq d_{\square}(x_j)\} \tag{8}$$

where $x_i \in U$, $x_j \in U$ and $\square$ = {p, μ, γ, δ, ω }.

Definition 9. For a derived consistent decision table $S_{\square}$, $A \subseteq C$, the core set $Core_{\square}$ is defined as

$$Core_{\square} = \{a \mid \exists m_{ij}^{\square} = \{a\}, m_{ij}^{\square} \in M_{\square}\} \tag{9}$$

where $\square$ = {p, μ, γ, δ, ω }.

Theorem 1. For a derived consistent decision table $S_{\square}$, $Core_{\omega}$ and $Core_{\mu}$ are the core attribute sets for relative indiscernibility relation reduct and distribution reduct, we have $Core_{\omega} \supseteq Core_{\mu}$.

Theorem 2. For a derived consistent decision table $S_{\square}$, $Core_{\mu}$, $Core_{\gamma}$ and $Core_{\delta}$ are the core attribute sets for distribution reduct, maximum distribution reduct and assignment reduct, we have $Core_{\mu} \supseteq Core_{\gamma}$ and $Core_{\mu} \supseteq Core_{\delta}$.

Theorem 3. For a derived consistent decision table $S_{\square}$, $Core_{\delta}$ and $Core_{p}$ are the core attribute sets for assignment reduct and Pawlak reduct, we have $Core_{\delta} \supseteq Core_{p}$.

Example 1: Table 1 provides a decision table S, where U={1, 2, 3, 4, 5, 6, 7} is the set of objects, C={ c_1, c_2 } is the set of condition attributes, and D={d} is the set of decision attribute.

Table 1. A Decision Table

U	c_1	c_2	d
1	1	1	1
2	1	0	0
3	1	0	1
4	1	0	1
5	0	0	0
6	0	0	1
7	0	0	0

From Table 1, we can acquire the core of maximum distribution reduct {c_1}. However, the core of positive reduct is {c_2}. Thus, the two core sets are irrelevant. As well, we can compute the core of assignment reduct {c_2}. The core sets of assignment reduct and maximum distribution reduct are also irrelevant.

As illustrated in the above theorems, we have the following conclusions.

Theorem 4. For an inconsistent decision table S, we have

1. $Core_\omega \supseteq Core_\mu \supseteq Core_\delta \supseteq Core_p$;
2. $Core_\omega \supseteq Core_\mu \supseteq Core_\gamma$.

Theorem 5. For a consistent decision table S, we have $Core_p = Core_\mu = Core_\gamma = Core_\delta = Core_\omega$.

3.2. The Relationships among Five Relative Reducts

From Definition 6, one can check that a Pawlak reduct can keep the positive region unchanged; an assignment reduct can keep the possible decision of any object x unchanged; a distribution reduct can keep the distribution of rough membership values of all objects in U unchanged; a maximum distribution reduct can keep the maximum decision classes of all objects in U unchanged, and a relative indiscernibility relation reduct can keep the decision partition of any object x unchanged. Since the boundary region exists for an inconsistent decision table, different partitions induced by each type of relative reduct are combined as shown in Figure 1. All partitions in the boundary region for Pawlak reduct is regarded as an equivalence class in Figure 1(a), while the partitions in the boundary region for relative indiscernibility relation reduct are the same as that of the original dataset in Figure 1(c) and (d). For assignment reduct, some partitions are combined if the possible decisions of two equivalence classes are the same in Figure 1(b'). For distribution reduct, some partitions are combined if the rough membership distribution values of two equivalence classes are the same in Figure 1(b). For maximum distribution reduct, some partitions are combined if the maximum decision classes of two equivalence classes are the same in Figure 1(b'').

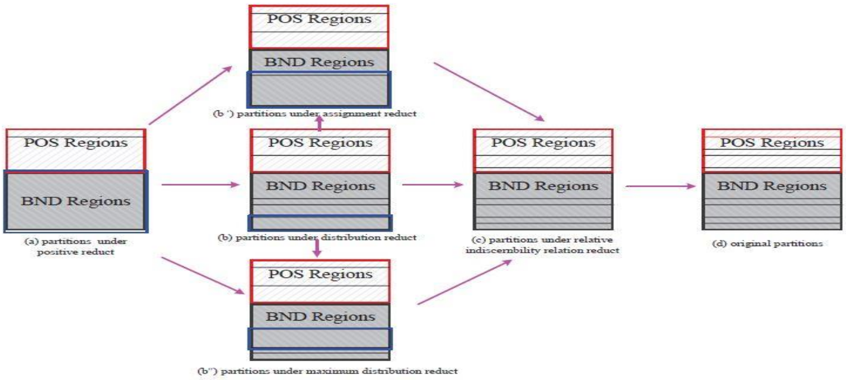

Figure 1. Different Partitions for Five Relative Reducts

In what follows, we study the relationships among five relative reducts from the algebra view, information view and relative discernibility view. The notation "1 $\Rightarrow$ 2" denotes that if 1 is satisfied, then 2 will be satisfied as well.

According to the available research results [2,29] and Theorems 1–3, we can obtain two theorems as follows.

Theorem 6. For an inconsistent decision table S, $\forall A \subseteq C$, $x \in U$, we have:

1. $[IND(A|D) = IND(C|D)] \Rightarrow [\mu_A(x) = \mu_C(x)] \Rightarrow [\gamma_A(x) = \gamma_C(x)]$;
2. $[IND(A|D) = IND(C|D)] \Rightarrow [\mu_A(x) = \mu_C(x)] \Rightarrow [\delta_A(x) = \delta_C(x)] \Rightarrow [POS_A(D) = POS_C(D)]$.

Theorem 7. For a consistent decision table S, $\forall A \subseteq C$, $x \in U$, we have: IND(A|D)=IND(C|D) $\Longleftrightarrow$ $[\mu_A(x) = \mu_C(x)]$ $\Longleftrightarrow$ $[\gamma_A(x) = \gamma_C(x)]$ $\Longleftrightarrow$ $[\delta_A(x) = \delta_C(x)]$ $\Longleftrightarrow$ $[POS_A(D) = POS_C(D)]$.

The relationships among five relative reducts are shown as in Figure 2.

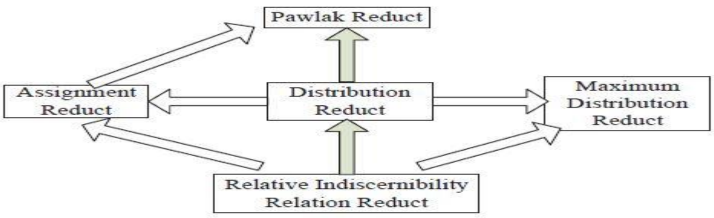

Figure 2. Relationships among the Superset-Subsets of Five Relative Reducts

4. Efficient Attribute Reduction Algorithms Using Relative Discernibility Degree for Inconsistent Decision Tables

As we all know, many researchers [3,16,22,23,24] employed discernibility matrix for attribute reduction. For small datasets, we can generate discernibility matrix to compute all reducts or a reduct. For large datasets, we cannot construct discernibility matrix for attribute reduction. Ngugen [12] and Korzeń [6] implemented efficient heuristic algorithms for acquiring a reduct through computing discernibility object pairs. Chen *et al.* [3] proposed an attribute reduction method using sample pair selection for all reducts. In order to further accelerate attribute reduction for large inconsistent decision tables without constructing discernibility matrix, we must design a more reasonable uncertainty measure to evaluate discernibility information. In this section, we discuss relative discernibility object pair and relative discernibility degree which is similar to the relative discernibility degree [19]. However, our uncertainty measure will generate less relative discernibility object pair and the computation is more feasible in parallel [17].

4.1. Discernibility Object Pair and Relative Discernibility Degree

Based on these studies, we define the discernibility object pair and relative discernibility degree, and design the efficient method for computing the number of discernibility object pairs.

Suppose π_D = $\{d_{\square}^{1}$, $d_{\square}^{2}$, …, $d_{\square}^{k}\}$ and π_A = $\{A_1$, A_2, …, $A_r\}$ for the derived consistent decision table $S_{\square}$. In A_p, there exist n_p^i objects whose decision value is *i*. Obviously, all objects whose decision value is i in any equivalence class forms $d_{\square}^{i}$. In other words, n_1^i + … + n_r^i equals n^i, the number of the objects in $d_{\square}^{i}$. Similarly, we can check that n_1^1 + … + n_r^1 + … + n_1^k + … + n_r^k equals *n*, the number of the objects in U. For any two objects, if the decision values are different and the combinational values on condition attributes A are also different, then A can discern that two objects.

Definition 10. For a derived consistent decision table $S_{\square}$, let $A \subseteq C$, π_D = $\{d_{\square}^{1}$, $d_{\square}^{2}$, …, $d_{\square}^{k}\}$, the discernibility object pair set of A with respect to $d_{\square}$ can be defined as $DOP_A^{\square}$:

$$DOP_A^{\square} = \{<x, y>| x \in d_{\square}^{i}, y \in d_{\square}^{j}, \exists a \in A, I_a(x) \neq I_a(y)\} \tag{10}$$

where $d_{\square}^{i} \in \pi_D$, $d_{\square}^{j} \in \pi_D$, $1 \le i < j \le k$.

The relative discernibility degree of A with respect to D is denoted by $|DOP_A^{\square}|$, which is equal to the number of discernibility object pairs. Thus, how to efficiently compute the number of discernibility object pairs is one of the most crucial issues in attribute reduction.

Definition 11. For a derived consistent decision table $S_{\square}$, let $A \subseteq C$ and $=\{A_1$, A_2, …, $A_r\}$, we get the number of object pairs that condition attributes A can discern.

$$DIS_A^{\square} = \sum_{1 \le i < j \le k} \sum_{1 \le p < q \le r} n_p^i n_q^j \tag{11}$$

According to the above equation, computing $DIS_A^{\square}$ (the relative discernibility degree) is very complex, therefore we calculate the number of those object pairs that A can not discern in turn. The main idea is that for any two objects, if the decision values are different and the combinational attribute values in terms of condition attributes A are the same, then A can not discern that two objects. An indiscernibility object pair with respect to the conditional attributes A is generated from any two objects which have different decision values and the same combinational values on A.

Definition 12. For a derived consistent decision table $S_{\square}$, let $A \subseteq C$, π_D = $\{d_{\square}^{1}$, $d_{\square}^{2}$, …, $d_{\square}^{k}\}$, the indiscernibility object pair set of A with respect to D can be defined as $DOP_A^{\square}$:

$$DOP_A^{\square} = \{<x, y>| x \in d_{\square}^{i}, y \in d_{\square}^{j}, \forall a \in A, I_a(x) = I_a(y)\} \tag{12}$$

where $d_{\square}^{i} \in \pi_D$, $d_{\square}^{j} \in \pi_D$, $1 \le i < j \le k$.

Definition 13. For a derived consistent decision table $S_{\square}$, let $A \subseteq C$ and $=\{A_1$, A_2, $\cdots$, $A_r\}$, we get the number of pairs of objects that condition attributes A can not discern.

$$DIS_A^{\square} = \sum_{1 \le p \le r} \sum_{1 \le i < j \le k} n_p^i n_p^j \tag{13}$$

According to Definition 13, if A is $\varnothing$, then $DIS_{\varnothing}^{\square} = \sum_{1 \le i < j \le k} n^i n^j$. When $DIS_A^{\square}$ does not equal 0, we must add some other attribute to discern those remaining indiscernibility object pairs. Assume that the selected attribute is an attribute *a*, then those object pairs are composed of some which attribute *a* can discern and the rest which attribute a can not discern.

Theorem 8. Given a derived consistent decision table $S_{\square}$ for $a \in C - A$, $DIS_{A \cup a}^{\square} + DIS_{A \cup a}^{\square} = DIS_A^{\square}$.

Theorem 9. Given a derived consistent decision table $S_{\square}$ for $a \in C$, $DIS_a^{\square}$ + $DIS_a^{\square}$ = $\sum_{1 \le i < j \le k} n^i n^j$.

According to Definitions 10 and 12, the computation of the relative discernibility degree will change into computing the number of indiscernibility object pairs. Since indiscernibility object pairs are generated from the boundary region with respect to condition attributes from A, our improved relative discernibility degree is more easily calculated, especially more suitable for parallel computing.

4.2. The Relationships among Three Attribute Reduction Algorithms Using Dependency Function, Information Gain and Relative Discernibility Degree

In what follows, we mainly discuss the relationships among three attribute reduction algorithms based on dependency function, information gain and relative discernibility degree.

Theorem 10. For a consistent decision table S, an attribute set $A \subseteq C$, we have [DIS_A^D = DIS_C^D = 0] $\Longleftrightarrow$ [$H(D|A)$ = $H(D|C)$ =0] $\Longleftrightarrow$ [$\gamma_A(D)$ = $\gamma_C(D)$ = 1].

Theorem 11. For an inconsistent decision table S, an attribute set $A \subseteq C$, if $\gamma_A(D)$ = $\gamma_C(D)$, then we have $H(D|A) \ge H(D|C)$ and $DIS_A^D \ge DIS_C^D$.

Theorem 12. For an inconsistent decision table S, an attribute set $A \subseteq C$, if $H(D|A)$ = $H(D|C)$, we can have $DIS_A^D \ge DIS_C^D$.

As indicated in Theorems 10-12, we can conclude that these three kinds of attribute reduction algorithms are inequivalent to each other for inconsistent decision tables but equivalent for consistent decision tables. In the following subsection, we mainly compute the equivalence classes, the derived consistent decision table and the core attributes using a counting sort algorithm, which plays a pivotal role in attribute reduction methods.

4.3. Efficient Attribute Reduction Algorithms from View of the Boundary Region

In rough set model, classical attribute reduction algorithms mainly deal with categorical data. Thus, we can recode the symbol attributes with a set of consecutive nature numbers. We mainly focus on discussing a decision table with a set of integral numbers in this paper. As we all know, an efficient heuristic attribute reduction method reflects in three aspects—computing equivalence classes, reducing storage and minimizing the search space. In order to accelerate the process of attribute reduction, we use an efficient counting sort algorithm for computing equivalence classes and core attribute, which time complexity is cut down to $O(|A||U|)$ [16]. In addition, we use our improved relative discernibility degree to construct an efficient attribute reduction framework from view of the boundary region which can turn into different attribute reduction algorithms to acquire different relative reducts.

Definition 14. For a derived consistent decision table $S_{\square}$, let $A \subseteq C$ and $a \in C - A$, then the significance of attribute a is defined by:

$$sig_{\square}(A, a, d_{\square}) = 1 - \frac{DIS^{\square}_{A \cup a}}{\sum_{1 \le i < j \le k} n^i n^j} \tag{14}$$

As $\sum_{1 \le i < j \le k} n^i n^j \ge DIS^{\square}_{A \cup a} \ge 0$, we have $0 \le sig_{\square}(A, a, d_{\square}) \le 1$ and this attribute measure is monotonic. Therefore, we can employ it to construct an efficient attribute reduction framework as shown in Algorithm 1.

Algorithm 1: Attribute reduction framework from view of the boundary region
Input: An inconsistent decision table S.
Output: A reduct Red.
Begin

1. Compute π_C and π_D using counting sort algorithm;
2. Compute the distribution of the decision attribute D and assign the values for new decision attribute $d_{\square}$ for each $C_i \in \pi_C$, and acquire the derived consistent decision table $S_{\square}$;
3. Let Red = $\varnothing$;
4. Sort the object sequence in terms of C, and compute $DIS^{\square}_{C-\{c_1\}}$; If $DIS^{\square}_{C-\{c_1\}}$ > 0, then $Core_{\square}(C)$ = $Core_{\square}(C) \cup \{c_1\}$;
5. For i=2 to m do

{ Sort the previous object sequence in terms of c_{i-1};
If $DIS^{\square}_{C-\{c_i\}}$ > 0, then $Core_{\square}(C)$ = $Core_{\square}(C) \cup \{c_i\}$;
}

6. Red = Red $\cup$ $Core_{\square}(C)$;
7. If $DIS^{\square}_{Red}$ = 0, then turn to Step 11;
8. To each attribute $a \in C - Red$, compute $DIS^{\square}_{Red \cup a}$;
9. Let $sig_{\square}(Red, a', d_{\square})$ = min ($sig_{\square}(Red, a, d_{\square})$)(if the attribute like that is not only one, select one attribute arbitrarily);
10. Red = Red $\cup a'$, turn to Step 8;
11. Output Red.

End.

The time complexity of step 1 is $O(|C||U|)$. That of Step 2 is $O(|C \cup D||U|)$. That of Step 3 is $O(1)$. The time complexity of Steps 4 and 5 is $O(|C||U|)$. The time complexity of Steps 6 and 7 is $O(1)$. The worst time complexity of Step 8 is $O(|C||U/C|)$. Thus the worst time complexity of the eighth step to the tenth step is $O(|U/C||C|)$ + $O(|U/C|(|C|-1))$ + $\cdots$ + $O(|U/C|)$ = $O(|C|^2|U/C|)$. Therefore, the time complexity of this algorithm is no more than max($O(|C||U|)$, $O(|C|^2|U/C|)$). The space complexity is no more than $O(|U|)$.

Example 2: Table 2 provides a decision table S, where U={1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11} is the set of objects, C={ c_1, c_2, c_3, c_4, c_5 } is the set of condition attributes, and D={d} is the set of decision attribute.

Table 2. A Decision Table

U	c_1	c_2	c_3	c_4	c_5	d
1	1	1	1	1	1	1
2	0	0	1	1	1	0
3	0	0	1	0	1	0
4	0	0	1	1	1	1
5	0	0	1	0	1	1
6	1	0	1	1	1	0
7	0	0	0	1	1	1
8	1	0	1	1	1	2
9	0	0	0	1	1	0
10	1	0	1	1	1	1
11	0	0	0	1	1	1

Here we can calculate the equivalence classes {3, 5}, {7, 9, 11},{2, 4}, {6, 8, 10} and {1} using counting sort algorithm. Thus, we easily acquire the derived consistent decision table and core attributes as Tables 3 and 4. Table 3 provides a derived consistent decision table S' from an inconsistent decision table S while Table 4 provides the core attribute set for a derived consistent decision table S'.

Table 3. A Derived Consistent Decision Table

No.	π_C	d_p	d_δ	d_γ	d_μ	d_ω	decision probability distribution
1	C_1 = {11111}	p_1	δ_1	γ_1	μ_1	ω_1	(0, 1, 0)
2	C_2 = {00111}	p_2	δ_2	γ_2	μ_2	ω_2	($\frac{1}{2}$, $\frac{1}{2}$, 0)
3	C_3 = {00101}	p_2	δ_2	γ_2	μ_2	ω_3	($\frac{1}{2}$, $\frac{1}{2}$, 0)
4	C_4 = {10111}	p_2	δ_3	γ_3	μ_3	ω_4	($\frac{1}{3}$, $\frac{1}{3}$, $\frac{1}{3}$)
5	C_5 = {00011}	p_2	δ_2	γ_1	μ_4	ω_5	($\frac{1}{3}$, $\frac{2}{3}$, 0)

Table 4. Number of Core Attribute Set for Five Reducts

$Core_p(C)$	$Core_\delta(C)$	$Core_\gamma(C)$	$Core_\mu(C)$	$Core_\omega(C)$
{ c_2 }	{ c_1, c_2 }	{ c_1, c_2, c_3 }	{ c_1, c_2, c_3 }	{ c_1, c_2, c_3, c_4 }

5. Experimental Analysis

5.1. Efficiency Evaluation

In order to evaluate the time efficiencies of our algorithms, we implement some experiments on a personal computer with Windows7, 2.40 GHZ CPU and 16GB memory. The software is Visual C# 2012. The objective of the following experimental results is to show the time efficiencies of attribute reduction algorithms using relative discernibility.

Since our approaches and traditional attribute reduction algorithms only deal with discrete attributes, we employ Rosetta software (http://www.lcb.uu.se/tools/rosetta/) to fill in some missing values and transform the numerical data into discrete ones. We perform the experiments on six datasets, which are three publicly available datasets from UCI Repository of machine learning databases in [1] and three synthetic datasets. Each dataset has only one decision attribute. For one synthetic datasets Dataset1, the values of their condition attributes and decision attribute are generated randomly from 0 to 9. For another two synthetic datasets Dataset2 and Dataset3, the values of their condition attributes and decision attribute are generated randomly from 1 to 5. The characteristics of six datasets are summarized in Table 5. Note that Mushroom* denotes the decision values of the objects are random.

Table 5. Description of the Datasets

No	Dataset	Objects	Attributes	Class	Consistency	Boundary Region
1	Chess-end-game	3196	36	2	yes	0
2	Mushroom*	8124	22	2	no	4012
3	Connect-4	67557	42	3	yes	0
4	Dataset1	500,000	30	10	no	87575
5	Dataset2	500,000	30	5	no	125035
6	Dataset3	1,000,000	30	5	no	204337

We here compare an attribute reduction algorithm based on discernibility matrix(DM) with our algorithm(DIS) as shown in Figure 3. Although both the running time of DM and DIS increase with the size of the datasets, the former consumes large memory and increases much more rapidly than the latter because DM algorithm needs to generate large discernibility matrix, especially when the number of the objects is greater than 10,000.

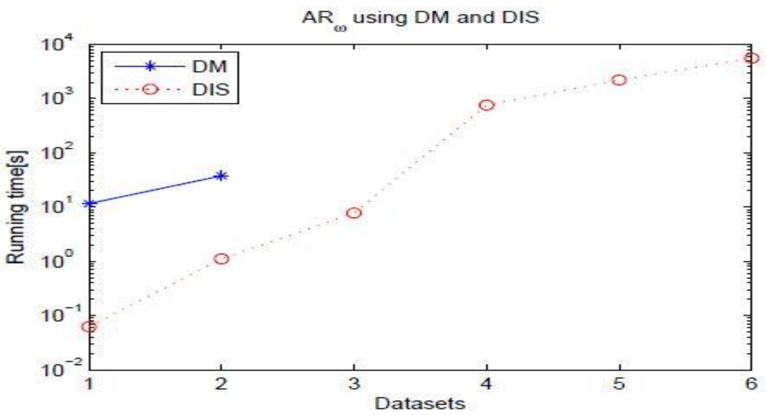

Figure 3. The Running Time for DM and DIS

In the following experiments, we mainly discuss running time, number of selected attributes and average length of decision rules of our algorithm using relative discernibility degree. We divided the dataset Connect-4 into 7 parts according to the number of objects. The first 10000 objects were regarded as the first dataset, the first 20000 objects were viewed as the second dataset, and so on. The whole objects were viewed as the seventh dataset. We divided the dataset Datasets6 to 10 parts according to the number of objects. As indicated in Figure 4 and 5, we compare running time, numbers of selected attribute and average length with respect to the number of increasing objects. One can check that the effect of the proposed relative discernibility degree measure is similar to that of information entropy. The running time for attribute reduction based on dependency function is shorter because the computation of dependency measure is simpler than that of the other two measures. However, the number of selected attribute is much bigger and average lengths of decision rules is much longer for dependency function for different incremental datasets of Connect-4 in Figure 4.

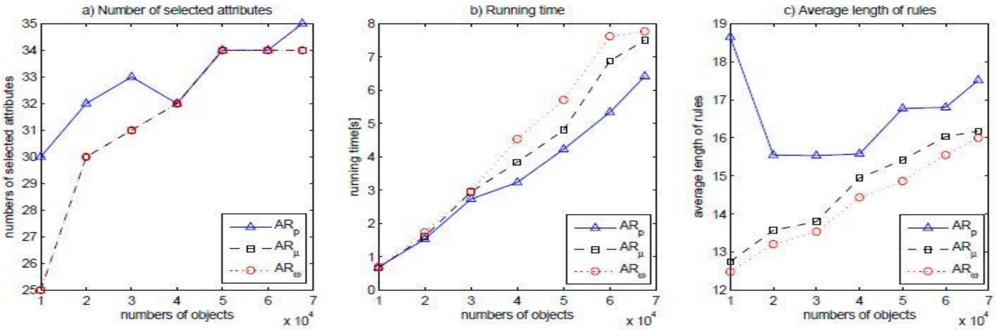

Figure 4. Number of Selected Attributes, Running Time and Average Length of Rules on Connect-4

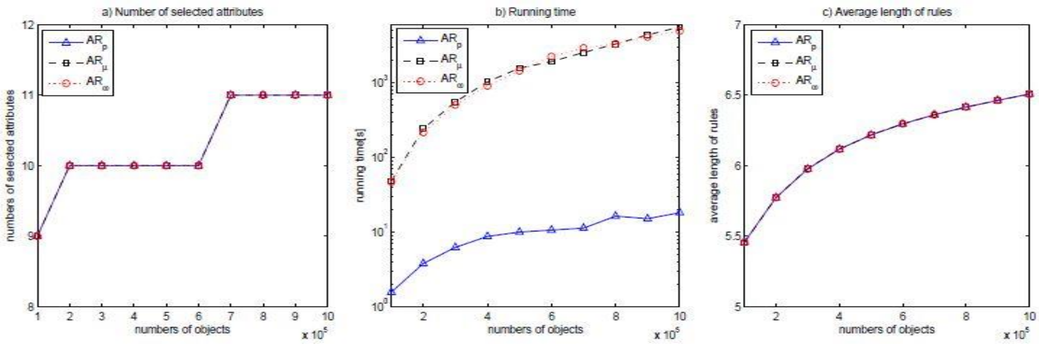

Figure 5. Number of Selected Attributes, Running Time and Average Length of Rules on Dataset6

Then, we compare the number of the derived decision classes in the boundary region and running time for five different relative reducts acquired by our algorithms in Figs. 6 and 7. We also compare our algorithm (Dis) with other two attribute reduction algorithms using dependency function (POS) and information entropy (Info) for five relative reducts on four inconsistent datasets in Figure 7. It is found that our algorithm is similar to the attribute reduction algorithm based on information entropy. One can also check that the running time of distribution reduct(AR_μ) and the relative indiscernibility relation reduct(AR_ω) are much longer than those of three other relative reducts. In general, the more the merged equivalence classes in the boundary region are, the longer the running time is.

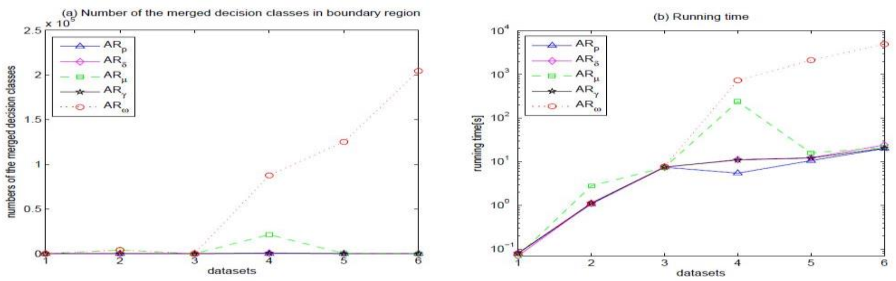

Figure 6. Number of Different Boundary Classes and Running Time on Six Datasets for Five Relative Reducts

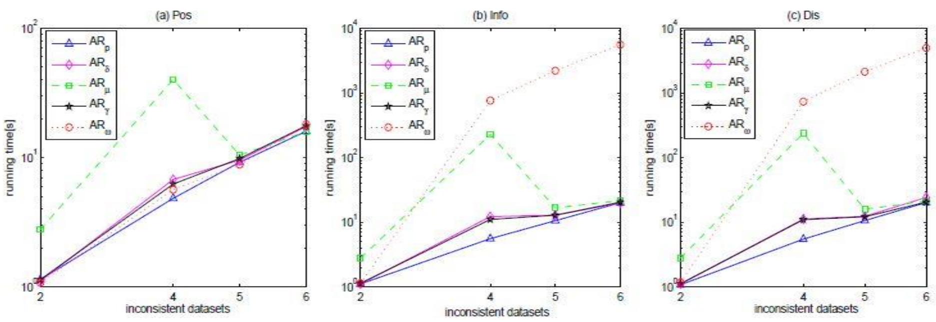

Figure 7. Running Time on Four Inconsistent Datasets for Three Algorithms

These differences can be illustrated as shown in Table 7 in details. The reason is that AR_{μ} and AR_{ω} need to discern much more equivalence classes in the boundary region. For example, they must discern 21238 and 87545 equivalence classes for Dataset 2. Although the sizes of Dataset1 and Dataset2 are almost the same, the running time for Dataset2 is much shorter than that for Dataset1 since the number of the equivalence classes of the inconsistent objects on Dataset2 is less than that on Dataset1 for AR_{ω}.

Table 7. Size of Boundary Classes and Length/Number for Five Reducts

Reducts	Mushroom*		Dataset1		Dataset2		Dataset3	
	size	$\frac{length}{number}$	size	$\frac{length}{number}$	size	$\frac{length}{number}$	size	$\frac{length}{number}$
AR_p	1	$\frac{86832}{6223}$	1	$\frac{321018}{58527}$	1	$\frac{954745}{120204}$	1	$\frac{1821826}{287061}$
AR_{δ}	5	$\frac{86863}{6225}$	502	$\frac{598948}{99786}$	26	$\frac{1373970}{165394}$	1	$\frac{1821826}{287061}$
AR_{μ}	4012	$\frac{101309}{7141}$	21238	$\frac{599150}{99812}$	226	$\frac{1381947}{166145}$	10	$\frac{2111053}{325115}$
AR_{γ}	153	$\frac{97378}{6890}$	502	$\frac{593750}{99087}$	26	$\frac{1351397}{163093}$	1	$\frac{1930696}{302217}$
AR_{ω}	4012	$\frac{101309}{7141}$	87545	$\frac{599197}{99818}$	125035	$\frac{1383758}{166353}$	204337	$\frac{2133390}{327859}$

5.2. Related Discussion

In this subsection, we summarize the advantages of our algorithms for attribute reduction and offer some comments. Nguyen in [12] proposed the algorithm for computing of positive regions in $O(|C||U|log|U|)$ time using $O(|U|)$ space. This algorithm used the lexicographical order and sorted objects in $O(|C||U|log|U|)$ time. However, we introduce a counting sort algorithm with time complexity $O(|C||U|)$ to sort objects for computing positive regions and core attributes. In addition, Nguyen implemented Johnson strategy for computing of short reducts in $O(|C|^2|U|log|U|)$, which is similar to algorithm 1 employing quick sort technique. But our algorithm takes time in $max(O(|C||U|),O(|C|^2|U/C|))$ without building the discernibility matrix as well. The main improvement in our algorithms lies in the sort technique, equivalence classes and a more reasonable uncertainty measure. From the experimental analysis, we can conclude that our approaches are faster than the corresponding algorithms [10,12,25] in general, especially for large inconsistent datasets. Furthermore, our algorithms can easily compute different definitions of relative reducts for the derived consistent decision tables.

6. Conclusions

Attribute reduction can reduce dimensionality and improve classification accuracy. In order to improve attribute reduction, we first discussed the relationships among different core attributes and different definitions of relative reducts, then designed the derived consistent decision table and a more reasonable uncertainty measure—relative discernibility degree, and finally proposed an efficient attribute reduction algorithm framework using discernibility degree with time complexity no more than $max(O(|C||U|), O(|C|^2|U/C|))$ for feature selection in data mining. A series of experiments are conducted on six datasets for evaluating the proposed algorithms. The results show that our methods are effective and can efficiently obtain different relative reducts.

Future work includes improving the proposed algorithm further and mining decision rules from large datasets.

Acknowledgments

The research is supported by the Humanity and Social Science Youth Foundation of Ministry of Education of China (No. 15YJCZH129), the Natural Science Foundation of Jiangsu Province (No. BK20141152), Qing Lan Project of Jiangsu Province of China, and Jiangsu Key Laboratory of Big Data Analysis Technology / B-DAT(Nanjing University of Information Science & Technology)(No. KXK1402).

References

[1] A. Asuncion and D. J. Newman, "UCI Machine Learning Repository [http://www.ics.uci.edu/ mlearn /MLRepository.html]", Irvine, CA: University of California, Department of Information and Computer Science, **(2007)**.

[2] Y. H. Chang, Q. J. Guo and J. Qian, "Quick Knowledge reduction algorithm in inconsistent information system", Journal of Chinese Computer Systems, vol. 36, no. 8, **(2015)**, pp. 1852-1856.

[3] D. G. Chen, S. Y. Zhao, L. Zhang, Y. P. Yang and X. Zhang, "Sample pair selection for attribute reduction with rough set", IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 11, **(2012)**, pp. 2080-2092.

[4] H. M. Chen, T. R. Li, C. Luo, S. J. Horng and G. Y. Wang, "A rough set-based method for updating decision rules on attribute values' coarsening and refining", IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 12, **(2014)**, pp. 2886-2899.

[5] Y. L. Jiang, Z. X. Yang and Y. Liu, "Quick distribution reduction algorithm in inconsistent information system", Acta Automatica Sinica, vol. 38, no. 3, **(2012)**, pp. 382-388.

[6] M. Korzeń and S. Jaroszewicz, "Finding reducts without building the discernibility matrix", in: Proceedings of 5th International Conference on Intelligent Systems Design and Applications (ISDA'05), **(2005)**.

[7] M. Kryszkiewicz, "Comparative study of alternative types of knowledge reduction in inconsistent systems", International Journal of Intelligent Systems, vol. 16, no. 1, **(2001)**, pp. 105-120.

[8] F. Li, Q. H. Liu, M. Ye and G. W. Yang, "Approaches to knowledge reductions in inconsistent decision tables", Control and Decision, vol. 21, no. 8, **(2006)**, pp. 857-862.

[9] M. Li, C. X. Shang, S. Z. Feng and J. P. Fan, "Quick attribute reduction in inconsistent decision tables", Information Sciences, vol. 254, **(2014)**, pp. 155-180.

[10] Q. H. Liu, F. Li, F. Min, M. Ye and G. W. Yang, "An efficient knowledge reduction algorithm based on new conditional information entropy", Control and Decision, vol. 20, **(2005)**, pp. 878-882.

[11] D. Q. Miao, Y. Zhao, Y. Y. Yao, H. X. Li and F. F. Xu, "Relative reducts in consistent and inconsistent decision tables of the Pawlak rough set model", Information Sciences, vol. 179, **(2009)**, pp. 4140-4150.

[12] S. H. Nguyen and H. S. Nguyen, "Some efficient algorithms for rough set methods", Proceedings of the International Conference on Information Processing and Management of Uncertainty on Knowledge Based Systems, **(1996)**, pp. 1451-1456.

[13] N. M. Partháln and Q. Shen, "Exploring the boundary region of tolerance rough sets for feature selection", Pattern Recognition, vol. 42, **(2009)**, pp. 655-667.

[14] Z. Pawlak, "Rough sets", International Journal of Computer and Information Sciences, vol. 11, **(1982)** pp. 341-356.

[15] Y. H. Qian, J. Y. Liang, W. Pedrycz and C. Y. Dang, "Positive approximation: An accelerator for attribute reduction in rough set theory", Artificial Intelligence, vol. 174, no. 9, **(2010)**, pp. 597-618.

[16] J. Qian, D. Q. Miao, Z. H. Zhang and W. Li, “Hybrid approaches to attribute reduction based on indiscernibility and discernibility relation”, International Journal of Approximate Reasoning, vol. 52, no. 2, **(2011)**, pp. 212-230.
[17] J. Qian, D. Q. Miao, Z. H. Zhang and X. D. Yue, “Parallel attribute reduction algorithms using mapreduce”, Information Sciences, vol. 279, **(2014)**, pp. 671-690.
[18] A. Skowron and C. Rauszer, “The discernibility matrices and functions in information systems, In: R. Slowiński (Ed.), Intelligent Decision Support”, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, **(1992)**.
[19] S. H. Teng, M. Lu, A. F. Yang, J. Zhang, Y. J. Nian and M. He, “Efficient attribute redution from the viewpoint of discernibility”, Information Sciences, vol. 326, **(2016)**, pp. 297-314.
[20] G. Y. Wang, H. Yu and D. C. Yang, “Decision table reduction based on conditional information entropy”, Chinese Journal of Computers, vol. 25, **(2002)**, pp. 760-766.
[21] Z. Y. Xu, Z. P. Liu, B. R. Yang and W. Song, “A quick attribute reduction algorithm with complexity of max($O(|C||U|), O(|C|^2|U/C|)$)”,Chinese Journal of Computers, vol. 29, no. 3, **(2006)**, pp. 611-615.
[22] M. Yang, “An incremental updating algorithm of the computation of a core based on the improved discernibility matrix”, Chinese Journal of Computers, vol. 29, no. 3, **(2006)**, pp. 407-413.
[23] Y. Y. Yao and Y. Zhao, “Discernibility matrix simplification for constructing attribute reducts”, Information Sciences, vol. 179, no. 7, **(2009)**, pp. 867-882.
[24] D. Y. Ye and Z. J. Chen, “A new type of attribute reduction for inconsistent decision tables and its computation”, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 18, no. 2, **(2010)**, pp. 209-222.
[25] W. X. Zhang, J. S. Mi and W. Z. Wu, “Approaches to knowledge reductions in inconsistent information systems”, International Journal of Intelligent Systems, vol. 18, no. 9, **(2003)**, pp. 989-1000.
[26] Y. Zhao, Y. Y. Yao and F. Luo, “Data analysis based on discernibility and indiscernibility”, Information Sciences, vol. 177, **(2007)**, pp. 4959-4976.
[27] N. Zhong and J. Z. Dong, “Using rough sets with heuristics for feature selection”, Journal of Intelligent Information Systems, vol. 16, **(2001)**, pp. 199-214.
[28] J. W. Zhou, B. Q. Feng and Y. Liu, “New algorithm for quick computing core”, Journal of Xi’an Jiaotong University, vol. 41, no. 6, **(2007)**, pp. 688-691.
[29] J. Zhou, D. Q. Miao, W. Pedrycz and H. Y. Zhang, “Analysis of alternative objective functions for attribute reduction in complete decision tables”, Soft Computing, vol. 15, **(2011)**, pp. 1601-1616.

Authors

Yuhui Chang. She received her M.Sc. in Computer Application (2007) from Yangzhou University. Her current research interests include Artificial Intelligence and Data Mining.