Attribute reduction for dynamic data sets

Feng Wang[a], Jiye Liang[a,*], Chuangyin Dang[b]

[a] Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, Shanxi, China
[b] Department of System Engineering and Engineering Management, City University of Hong Kong, Hong Kong, Hong Kong

ARTICLE INFO

ABSTRACT

Many real data sets in databases may vary dynamically. With such data sets, one has to run a knowledge acquisition algorithm repeatedly in order to acquire new knowledge. This is a very time-consuming process. To overcome this deficiency, several approaches have been developed to deal with dynamic databases. They mainly address knowledge updating from three aspects: the expansion of data, the increasing number of attributes and the variation of data values. This paper focuses on attribute reduction for data sets with dynamically varying data values. Information entropy is a common measure of uncertainty and has been widely used to construct attribute reduction algorithms. Based on three representative entropies, this paper develops an attribute reduction algorithm for data sets with dynamically varying data values. When a part of data in a given data set is replaced by some new data, compared with the classic reduction algorithms based on the three entropies, the developed algorithm can find a new reduct in a much shorter time. Experiments on six data sets downloaded from UCI show that the algorithm is effective and efficient.

© 2012 Elsevier B.V. All rights reserved.

1. Introduction

In real world databases, data sets usually vary with time. This phenomenon occurs in many fields such as economic research, social survey and medical research. As data sets change with time, especially at an unprecedented rate, it is very time-consuming or even infeasible to run repeatedly a knowledge acquisition algorithm. To overcome this deficiency, researchers have recently proposed many new analytic techniques. They usually can directly carry out the computation using the existing result from the original data set. These techniques mainly address knowledge updating from three aspects: the expansion of data [1–7], the increasing number of attributes [8–11] and the variation of data values [12,13]. For the first two aspects, a number of incremental techniques have been developed to acquire new knowledge without recomputation. However, little research has been done on the third aspect in knowledge acquisition, which motivates this study. This paper concerns attribute reduction for data sets with dynamically varying data values. For convenience of the following discussion, here are some specific explanations regarding data sets with dynamically varying data values. Generally speaking, this case usually occurs in the following several situations. One situation is that a part of the original data in a database is identified as wrong, thus needing to be corrected. Wrong data obviously lose their value to store further, and will be directly replaced by correct ones. Another familiar situation is that, initially collected data in a database may increase gradually, though it usually is not necessary to acquire knowledge from total data all the time. In other words, the sizes of interested data sets do not change. For example, in a pollution survey of X city, observed data in last few years or even decades may be stored in a database totally. However, analysis of pollution in each week (or each day, each month, etc.) does not require total observed data in the past. In this situation, because data sets of adjacent time intervals are usually similar to each other, an interested data set at one moment can be slightly amended to obtain a data set for the next moment. In addition, with the rapid development of information technology, timeliness of data becomes more and more important. Thus, any out-of-date data in databases are usually useless. To improve the efficiency of knowledge acquisition, useless data should be directly updated by the latest or real-time ones. Furthermore, other similar situations occur rather often in applications such as stock analysis, tests for the disease and annual appraisal of workers.

Feature selection, a common technique for data preprocessing in many areas including pattern recognition, machine learning and data mining, has hold great significance. Among various approaches to select useful features, a special theoretical framework is Pawlak's

---

rough set model [14,15]. One can use rough set theory to select a subset of features that is most suitable for a given recognition problem [16–21]. Rough feature selection is also called attribute reduction, which aims to select those features that keep the discerniblity ability of the original ones [22–26]. The feature subset generated by an attribute reduction algorithm is called a reduct. In the last two decades, researchers have proposed many reduction algorithms [27–32]. However, most of these algorithms can only be applicable to static data sets. Although several algorithms have been proposed for dynamic data sets [1–11], as mentioned above, they are incremental approaches only for the dynamic expansion of data or attributes.

This paper focus on attribute reduction for dynamically varying data values. To tackle this problem, this paper will exploit information entropy. The information entropy from classical thermodynamics is used to measure out-of-order degree of a system. Information entropy is introduced in rough set theory to measure uncertainty of a given data set [33–36], which have been widely applied to devise heuristic attribute reduction algorithms [37–41]. Complementary entropy [33], combination entropy [35] and Shannon's entropy [36] are three representative entropies which have been mainly used to construct reduction algorithms in rough set theory. To fully explore properties in reduct updating caused by the variation of data values, this paper develops an attribute reduction algorithm for dynamic data sets based on the three entropies. In view of that a key step of the development is the computation of entropy, this paper first introduces three updating mechanisms of the three entropies, which determine an entropy by changing one object to a new one in a decision table. When only one object is changed, instead of recomputation on the given decision table, the updating mechanisms derive new entropies by integrating the changes of conditional classes and decision classes into the existing entropies. With these mechanisms, an attribute reduction algorithm is proposed for dynamic decision tables. When a part of data in a given data set is replaced by some new data, compared with the classic reduction algorithms based on the three entropies, the developed algorithm can find a new reduct in a much shorter time. Experiments on six data sets downloaded from UCI show that the algorithm is effective and efficient.

The rest of this paper is organized as follows. Some preliminaries in rough set theory are briefly reviewed in Section 2. Traditional heuristic reduction algorithms based on three representative entropies are introduced in Section 3. Section 4 presents the updating mechanisms of the three entropies for dynamically varying data values. In Section 5, based on the updating mechanisms, a reduction algorithm is proposed to compute reducts for dynamic data sets. In Section 6, six UCI data sets are employed to demonstrate effectiveness and efficiency of the proposed algorithm. Section 7 concludes this paper with some discussions.

2. Preliminary knowledge on rough sets

2.1. Basic concepts

This section reviews several basic concepts in rough set theory. Throughout this paper, the universe U is assumed a finite nonempty set.

An information system, as a basic concept in rough set theory, provides a convenient framework for the representation of objects in terms of their attribute values. An information system is a quadruple $S = (U, A, V, f)$, where U is a finite nonempty set of objects and is called the universe and A is a finite nonempty set of attributes, $V = \bigcup_{a \in A} V_a$ with V_a being the domain of a, and $f : U \times A \to V$ is an information function with $f(x, a) \in V_a$ for each $a \in A$ and $x \in U$. The system S can often be simplified as $S = (U, A)$.

Each nonempty subset $B \subseteq A$ determines an indiscernibility relation in the following way,

$$R_B = \{(x, y) \in U \times U \mid f(x, a) = f(y, a), \forall a \in B\}.$$

The relation R_B partitions U into some equivalence classes given by

$$U/R_B = \{[x]_B \mid x \in U\}, \text{ just } U/B,$$

where $[x]_B$ denotes the equivalence class determined by x with respect to B, i.e.,

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

Given an equivalence relation R on the universe U and a subset $X \subseteq U$, one can define a lower approximation of X and an upper approximation of X by

$$\underline{R}X = \bigcup \{x \in U \mid [x]_R \subseteq X\}$$

and

$$\overline{R}X = \bigcup \{x \in U \mid [x]_R \cap X \neq \emptyset\},$$

respectively [39]. The order pair $(\underline{R}X, \overline{R}X)$ is called a rough set of X with respect to R. The positive region of X is denoted by $POS_R(X) = \underline{R}X$.

A partial relation $\preceq$ on the family $\{U/B \mid B \subseteq A\}$ is defined as follows [37]: $U/P \preceq U/Q$ (or $U/Q \succeq U/P$) if and only if, for every $P_i \in U/P$, there exists $Q_j \in U/Q$ such that $P_i \subseteq Q_j$, where $U/P = \{P_1, P_2, \ldots, P_m\}$ and $U/Q = \{Q_1, Q_2, \ldots, Q_n\}$ are partitions induced by $P, Q \subseteq A$, respectively. In this case, we say that Q is coarser than P, or P is finer than Q. If $U/P \preceq U/Q$ and $U/P \neq U/Q$, we say Q is strictly coarser than P (or P is strictly finer than Q), denoted by $U/P \prec U/Q$ (or $U/Q \succ U/P$).

It is clear that $U/P \prec U/Q$ if and only if, for every $X \in U/P$, there exists $Y \in U/Q$ such that $X \subseteq Y$, and there exist $X_0 \in U/P$ and $Y_0 \in U/Q$ such that $X_0 \subset Y_0$.

A decision table is an information system $S = (U, C \cup D)$ with $C \cap D = \emptyset$, where an element of C is called a condition attribute, C is called a condition attribute set, an element of D is called a decision attribute, and D is called a decision attribute set. Given $P \subseteq C$ and $U/D = \{D_1, D_2, \ldots, D_r\}$, the positive region of D with respect to the condition attribute set P is defined by $POS_P(D) = \bigcup_{k=1}^{r} \underline{P}D_k$.

For a decision table S and $P \subseteq C$, $X \in U/P$ is consistent iff all its objects have the same decision value; otherwise, X is inconsistent. The decision table S is called a consistent decision table iff $\forall X \in U/C$ are consistent; and if $\exists x, y \in U$ are inconsistent, then the table is called an inconsistent decision table. One can extract certain decision rules from a consistent decision table and uncertain decision rules from an inconsistent decision table.

For a decision table S and $P \subseteq C$, when a new object x is added to S, x is indistinguishable on B iff, $\exists y \in U$, $\forall a \in P$, such that $f(x, a) = f(y, a)$; and x is distinguishable on P iff, $\forall y \in U$, $\exists a \in P$ such that $f(x, a) \neq f(y, a)$.

2.2. Attribute reduction in rough set theory

Given an information system, all the attributes are not necessarily in the same importance, and some of them are irrelevant to the learning or recognition tasks. The concept of attribute reduction was first originated by Pawlak in [14,15], which aimed to delete the irrelevant or redundant attributes on the condition of retaining the discernible ability of original attributes (the whole attributes set). The retained attribute subset got from attribute reduction is called a reduct.

Definition 1. Let $S = \{U, A\}$ be an information system. Then $B \subseteq A$ is a reduct of S if

(1) $U/B = U/A$ and
(2) $\forall a \in B$, $U/(B - \{a\}) \neq U/B$.

There are usually multiple reducts in a given information system, and the intersection of all reducts is called core. Given a decision table, the retained attribute subset got from attribute reduction is called a relative reduct, and the intersection of all relative reducts is called relative core [14,15].

Definition 2. Let $S = \{U, C \cup D\}$ be a decision table. Then $B \subseteq C$ is a relative reduct of S if

(1) $POS_B(D) = POS_C(D)$ and
(2) $\forall a \in B$, $POS_{B-\{a\}}(D) \neq POS_B(D)$.

For these two definitions, the first condition guarantees that the reduct has the same distinguish power as the whole attribute set, and the second condition guarantees that there is no redundant attributes in the reduct. A reduct is called an exact reduct if it satisfies both of these two constraints, otherwise, is just an approximate reduct. In [40], Skowron proposed a discernibility matrix method to find all exact reducts of an information system without decision attributes. However, it has been proved that using this algorithm to generate reducts is an NP-hard problem. For decision tables, Kryszkiewicz proposed an approach to computing the minimal set of attributes that functionally determine a decision attribute [41]. This algorithm can find an exact reduct of a given decision table. These two approaches are both very time-consuming.

As is well known, Pawlak's rough set model is applicable for the case that only nominal attributes exist in data sets. However, many real data in applications usually come with a complicated form. To conceptualize and analyze various types of data, researchers have generalized Pawlak's classic rough set model, and attribute reduction based on these generalizations was also redefined. Reducts generated by these reduction algorithms are usually approximate reducts. Ziarko provided the concept of β-reduct based on the introduction of variable precision rough set model (VPRS) [42]. VPRS deals with partial classification by introducing a probability value β. The β value represents a bound on the conditional probability of objects in a condition class which are classified to the same decision class. Yao proposed the decision-theoretic rough set model and also defined attribute reduction based on this generalized model [43,44]. This model with loss functions aims to obtain optimization decisions by minimizing the overall risk with Bayesian decision procedures. An extensive review of multi-criteria decision analysis based on dominance rough sets was given by Greco et al. [45]. Dominance rough set model has also been applied for ordinal attribute reduction and multi-criteria classification [46,47]. Dubois and Prade constructed the first fuzzy rough model by extending equivalence relation to fuzzy equivalence relation [48], where fuzzy equivalence relations satisfy reflexivity, symmetry and max-min transitivity. Reduction algorithms based on above generalized rough set models usually generate one or more approximation reducts and have been applied to solve their corresponding issues. It is deserved to point out that each kind of attribute reduction tries to preserve a particular property of a given table.

In addition, to save computational time of finding reduct, researchers have also developed many heuristic attribute reduction algorithms which can generate a single reduct from a given table [33,35,37,38,49]. Most of them are greedy and forward search algorithms, keeping selecting attributes with high significance until the dependence no longer increases. The reduct generated by a heuristic reduction algorithm is usually considered as an approximation reduct. It will be an exact reduct when deleting its redundant attributes.

However, the above analysis about generating an exact reduct and an approximation reduct is not very rigorous. For example, though reducts generated by heuristic reduction algorithms are considered as approximation reducts, some of them are often exact redcuts. Because so many reduction algorithms have been proposed in the last two decades and it is very difficult to list all of them here, this section just introduces a common distinction between generating an exact reduct and generating an approximation reduct.

3. Attribute reduction based on information entropy

Among various heuristic attribute reduction algorithms, reduction based on information entropy (or its variants) is a kind of common algorithm which has attracted much attention. There are three representative entropies which are used to construct reduction algorithms. They are complementary entropy [33], combination entropy [35] and Shannon's information entropy [36]. The heuristic attribute reduction algorithms based on these three entropies are reviewed in this section.

In [33], the complementary entropy was introduced to measure uncertainty in rough set theory. Liang et al. also proposed the conditional complementary entropy to measure uncertainty of a decision table in [34]. By preserving the conditional entropy unchanged, the conditional complementary entropy was applied to construct reduction algorithms and reduce the redundant features in a decision table [35]. The conditional complementary entropy used in this algorithm is defined as follows [33–35].

Definition 3. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. Then, one can obtain the condition partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. Based on these partitions, a conditional entropy of B relative to D is defined as

$$E(D|B) = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{|X_i \cap Y_j|}{|U|} \frac{|Y_j^c - X_i^c|}{|U|}, \tag{1}$$

where Y_i^c and X_j^c are complement set of Y_i and X_j respectively.

Another information entropy, called combination entropy, was presented in [35] to measure the uncertainty of data tables. The conditional combination entropy was also introduced and can be used to construct the heuristic reduction algorithms [35]. This reduction algorithm can find a feature subset that possesses the same number of pairs of indistinguishable elements as that of the original decision table. The definition of the conditional combination entropy is defined as follows [35].

Definition 4. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. Then one can obtain the condition partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. Based on these partitions, a conditional entropy of B relative to D is defined as

$$CE(D|B) = \sum_{i=1}^{m} \left(\frac{|X_i|}{|U|} \frac{C_{|X_i|}^2}{C_{|U|}^2} - \sum_{j=1}^{n} \frac{|X_i \cap Y_j|}{|U|} \frac{C_{|X_i \cap Y_j|}^2}{C_{|U|}^2} \right). \tag{2}$$

where $C_{|X_i|}^2$ denotes the number of pairs of objects which are not distinguishable from each other in the equivalence class X_i.

Based on the classical rough set model, Shannon's information entropy [36] and its conditional entropy were also introduced to find a reduct in a heuristic algorithm [38,50]. In [38], the reduction algorithm keeps the conditional entropy of target decision unchanged, and the conditional entropy is defined as follows [38].

Definition 5. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. Then, one can obtain the condition partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. Based on these partitions, a conditional entropy of B relative to D is defined as

$$H(D|B) = -\sum_{i=1}^{m} \frac{|X_i|}{|U|} \sum_{j=1}^{n} \frac{|X_i \cap Y_j|}{|X_i|} \log \left(\frac{|X_i \cap Y_j|}{|X_i|} \right). \tag{3}$$

For convenience, a uniform notation $ME(D|B)$ is introduced to denote these three entropies. For example, if one adopts Shannon's conditional entropy to define the attribute significance, then $ME(D|B) = H(D|B)$. Given a decision table $S = (U, C \cup D)$ and $B_1, B_2 \subseteq C$. According to literatures [33,35,38], if $U/B_1 \preceq U/B_2$, one can get that $ME(D|B_1) \leq ME(D|B_2)$. This conclusion indicates that, for a given decision table, as its condition classifications become finer, its entropies (the three entropies) are monotone decreasing. In addition, as the condition classifications become finer, the classified quality (see Definition 11) of the given decision table is monotone increasing. Thus, one can get that the three entropies of a given decision table are monotone decreasing with the classified quality increasing. Especially, when the classified quality is one, the entropies are zero [33,35,38].

The attribute significances based on entropies in a heuristic reduction algorithm is defined as follows (see Definitions 6 and 7) [33,35,38].

Definition 6. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. $\forall a \in B$, the significance measure (inner significance) of a in B is defined as

$$Sig^{inner}(a, B, D) = ME(D|B - \{a\}) - ME(D|B). \tag{4}$$

Definition 7. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. $\forall a \in C - B$, the significance measure (outer significance) of a in B is defined as

$$Sig^{outer}(a, B, D) = ME(D|B) - ME(D|B \cup \{a\}). \tag{5}$$

Given a decision table $S = (U, C \cup D)$ and $a \in C$. From the literatures [26–29], one can get that if $Sig^{inner}(a, C, D) > 0$, then the attribute a is indispensable, i.e., a is a core attribute of S. Based on core attributes, a heuristic attribute reduction algorithm can find a reduct by gradually adding selected attributes to the core. The definition of reduct based on information entropy is defined as follows [26–29].

Definition 8. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. Then the attribute set B is a relative reduct if B satisfies:

(1) $ME(D|B) = ME(D|C)$;
(2) $\forall a \in B, ME(D|B) \neq ME(D|B - \{a\})$.

Formally, the searching strategies in reduction algorithms based on the three entropies are similar to each other. The specific steps are written as follows [33,35,38].

Algorithm 1. Classic attribute reduction algorithm based on information entropy for a decision table (CAR).

Input: A decision table $S = (U, C \cup D)$
Output: Reduct *red*
Step 1: $red \leftarrow \emptyset$;
Step 2: *for* $(j = 1; j \leq |C|; j++)$
{ if $Sig^{inner}(a_j, C, D) > 0$, then $red \leftarrow red \cup \{a_j\}$;
}
Step 4: $P \leftarrow red$, while $(ME(D|P) \neq ME(D|C))$ *do*

{

Compute and select sequentially $Sig^{outer}(a_0, P, D) = max\{Sig^{outer}(a_i, P, D)\}$, $a_i \in C - P$;

$P \leftarrow P \cup \{a_0\}$;

}

Step 5: $red \leftarrow P$, return red and end.

Based on Definition 6, one can get core attributes according to steps 1 and 2 in this algorithm. Steps 3 and 4 add selected attributes to the core gradually, and then one can obtain a reduct of the given table. This algorithm can be considered as the common attribute reduction algorithm based on information entropy. The time complexity of CAR given in [37] is $O(|U||C|^2)$. However, this time complexity does not include the computational time of entropies. For a given decision table, computing entropies is a key step in above reduction algorithm, which is not computationally costless. Thus, to analyze the exact time complexity of above algorithm, the time complexity of computing entropies is given as well.

Given a decision table, according to Definitions 3–5, it first needs to compute the conditional classes and decision classes, respectively, and then computes the value of entropy. Xu et al. in [51] gave a fast algorithm for partition with time complexity being $O(|U||C|)$. So, the time complexity of computing entropy is

$$
O\left(|U||C| + |U| + \sum_{i=1}^{m}|X_i| \cdot \sum_{j=1}^{n}|Y_j|\right) = O(|U||C| + |U| + |U||U|) = O(|U|^2),
$$

where the specific introduction of m, n, X_i and Y_j is shown in Definitions 3–5. Thus, the time complexity of computing core (steps 1 and 2) is $O(|C||U|^2)$, and the time complexity of computing reduct according to CAR is

$$
O(|C||U|^2 + |C|(|U||C| + |U|^2)) = O(|C|^2|U| + |C||U|^2).
$$

4. Updating mechanism of information entropy

Given a dynamic decision table, based on the three representative entropies, this section presents the updating mechanisms of the three entropies for dynamically varying data values. As data values in a decision table vary with time, recomputing entropy is obviously time-consuming. To overcome this deficiency, the updating mechanisms derive new entropies by integrating the changes of conditional classes and decision classes into existing entropies. When data values of a single object vary, Theorems 1–4 introduce the updating mechanisms for the three entropies respectively.

For convenience, here are some explanations which will be used in the following theorems. Let $S = (U, C \cup D)$ be a decision table, $B \subseteq C$ and $x \in U$. $U/B = \{X_1, X_2, \ldots, X_m\}$, $U/D = \{Y_1, Y_2, \ldots, Y_n\}$, $x \in X_{p_1}$ and $x \in Y_{q_1}$ ($p_1 \in \{1, 2, \ldots, m\}$ and $q_1 \in \{1, 2, \ldots, n\}$). If attribute values of x are varied, and here assumes that x is changed to x'. Let $U_{x'}$ denotes the new universe, one has $x' \in X'_{p_2}$ and $x' \in Y'_{q_2}$ ($X'_{p_2} \in U_{x'}/B$ and $Y'_{q_2} \in U_{x'}/D$). Obviously, one can get that $X'_{p_2} - \{x\} \in U/B$, $Y'_{q_2} - \{x\} \in U/D$, $X_{p_1} - \{x\} \in U_{x'}/B$ and $Y_{q_1} - \{x\} \in U_{x'}/D$. In addition, X'_{p_1}, Y'_{q_1}, X_{p_2} and Y_{q_2} denote $X_{p_1} - \{x\}$, $Y_{q_1} - \{x\}$, $X'_{p_2} - \{x\}$ and $Y'_{q_2} - \{x\}$, respectively.

Theorem 1. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. The complementary conditional entropy of D with respect to B is $E_U(D|B)$. Then, one can obtain the partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. $x \in X_{p_1}$ and $x \in Y_{q_1}$. If one and only object $x \in U$ is changed to x', then $x' \in X'_{p_2}$ and $x' \in Y'_{q_2}$ ($X'_{p_2} \in U_{x'}/B$ and $Y'_{q_2} \in U_{x'}/D$). The new conditional complementary entropy becomes

$$
E_{U_{x'}}(D|B) = E_U(D|B) + \frac{2|X'_{p_2} - Y'_{q_2}| - 2|X'_{p_1} - Y'_{q_1}|}{|U|^2},
$$

where $X'_{p_1} = X_{p_1} - \{x\}$ and $Y'_{q_1} = Y_{q_1} - \{x\}$.

Proof. For the decision table S, when x is changed to x', there are four situations about the changes of conditional classes and decision classes, which are as follows:

(a) $U_{x'}/B = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X_m, \{x'\}\}$ and $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y_n, \{x'\}\}$;

(b) $U_{x'}/B = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X'_{p_2}, \ldots, X_m, \}$, $x' \in X'_{p_2}$, and $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y_n, \{x'\}\}$;

(c) $U_{x'}/B = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X_m, \{x'\}\}$, $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y'_{q_2}, \ldots, Y_n\}$ and $x' \in Y'_{q_2}$;

(d) $U_{x'}/B = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X'_{p_2}, \ldots, X_m, \}$, $x' \in X'_{p_2}$ and $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y'_{q_2}, \ldots, Y_n\}$, $x' \in Y'_{q_2}$.

For convenience, here introduces a uniform notation about these four situations.

Let $U/B = \{X_1, X_2, \ldots, X_m, X_{m+1}\}$ and $X_{m+1} = \emptyset$. Then, we have $U_{x'}/B = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X'_{p_2}, \ldots, X_{m+1}, \}$, $X'_{p_1} = X_{p_1} - \{x\}$ and $X'_{p_2} = \{x'\} \cup X_{p_2}$. Similarly, let $Y_{n+1} = \emptyset$, we can get that $U/D = \{Y_1, Y_2, \ldots, Y_{n+1}\}$, $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y'_{q_2}, \ldots, Y_{n+1}\}$ and $Y'_{q_2} = \{x'\} \cup Y_{q_2}$. Obviously, for situation (a), we have $X'_{p_2} = X_{m+1} \cup \{x'\} = \emptyset \cup \{x'\} = \{x'\}$ and $Y'_{q_2} = Y_{n+1} \cup \{x'\} = \{x'\}$. Similarly, for situation (b), we have $Y'_{q_2} = Y_{n+1} \cup \{x'\} = \{x'\}$. And for situation (c), we have $X'_{p_2} = \{x'\}$.

According to the uniform notation, the updating mechanism of complementary conditional entropy is as follows.

$$
E(D|B) = \sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|X_i \cap Y_j|}{|U|}\frac{|Y_j^c - X_i^c|}{|U|} = \sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|X_i \cap Y_j|}{|U|}\frac{|X_i - Y_j|}{|U|}.
$$

Hence

$$E_{U_{x'}}(D|B) = \sum_{i=1}^{m-1} \left(\sum_{j=1}^{n-1} \frac{|X_i \cap Y_j||X_i - Y_j|}{|U|^2} + \frac{|X_i \cap Y'_{q_1}||X_i - Y'_{q_1}|}{|U|^2} + \frac{|X_i \cap Y'_{q_2}||X_i - Y'_{q_2}|}{|U|^2} \right) + \sum_{j=1}^{n-1} \frac{|X'_{p_1} \cap Y_j||X'_{p_1} - Y_j|}{|U|^2} + \frac{|X'_{p_1} \cap Y'_{q_1}||X'_{p_1} - Y'_{q_1}|}{|U|^2}$$

$$+ \frac{|X'_{p_1} \cap Y'_{q_2}||X'_{p_1} - Y'_{q_2}|}{|U|^2} + \sum_{j=1}^{n-1} \frac{|X'_{p_2} \cap Y_j||X'_{p_2} - Y_j|}{|U|^2} + \frac{|X'_{p_2} \cap Y'_{q_1}||X'_{p_2} - Y'_{q_1}|}{|U|^2} + \frac{|X'_{p_2} \cap Y'_{q_2}||X'_{p_2} - Y'_{q_2}|}{|U|^2}$$

$$= \sum_{i=1}^{m-1} \left(\sum_{j=1}^{n-1} \frac{|X_i \cap Y_j||X_i - Y_j|}{|U|^2} + \frac{|X_i \cap Y_{q_1}||X_i - Y_{q_1}|}{|U|^2} + \frac{|X_i \cap Y_{q_2}||X_i - Y_{q_2}|}{|U|^2} \right)$$

$$+ \sum_{j=1}^{n-1} \frac{|X_{p_1} \cap Y_j| \cdot (|X_{p_1} - Y_j| - 1)}{|U|^2} + \frac{(|X_{p_1} \cap Y_{q_1}| - 1) \cdot |X_{p_1} - Y_{q_1}|}{|U|^2} + \frac{|X_{p_1} \cap Y_{q_2}| \cdot (|X_{p_1} - Y_{q_2}| - 1)}{|U|^2} + \sum_{j=1}^{n-1} \frac{|X_{p_2} \cap Y_j| \cdot (|X_{p_2} - Y_j| + 1)}{|U|^2}$$

$$+ \frac{|X_{p_2} \cap Y_{q_1}| \cdot (|X_{p_2} - Y_{q_1}| + 1)}{|U|^2} + \frac{(|X_{p_2} \cap Y_{q_2}| + 1) \cdot |X_{p_2} - Y_{q_2}|}{|U|^2} = \sum_{i=1}^{m+1} \sum_{j=1}^{n+1} \frac{|X_i \cap Y_j||X_i - Y_j|}{|U|^2} + \frac{|X_{p_2} - Y_{q_2}| - |X_{p_1} - Y_{q_1}|}{|U|^2}$$

$$+ \sum_{j=1, j \neq q_2}^{n+1} \frac{|X_{p_2} \cap Y_j|}{|U|^2} - \sum_{j=1, j \neq q_1}^{n+1} \frac{|X_{p_1} \cap Y_j|}{|U|^2} = E_U(D|B) + \frac{|X'_{p_2} - Y'_{q_2}| - |X'_{p_1} - Y'_{q_1}|}{|U|^2} + \sum_{j=1, j \neq q_2}^{n+1} \frac{|X'_{p_2} \cap Y_j|}{|U|^2} - \sum_{j=1, j \neq q_1}^{n+1} \frac{|X'_{p_1} \cap Y_j|}{|U|^2} = E_U(D|B)$$

$$+ \frac{|X'_{p_2} - Y'_{q_2}| - |X'_{p_1} - Y'_{q_1}|}{|U|^2} + \sum_{j=1, j \neq q_2}^{n+1} \frac{|X'_{p_2} \cap Y_j|}{|U|^2} + \frac{|X'_{p_2} \cap Y'_{q_2}|}{|U|^2} - \frac{|X'_{p_2} \cap Y'_{q_2}|}{|U|^2} - \sum_{j=1, j \neq q_1}^{n+1} \frac{|X'_{p_1} \cap Y_j|}{|U|^2} - \frac{|X'_{p_1} \cap Y'_{q_1}|}{|U|^2} + \frac{|X'_{p_1} \cap Y'_{q_1}|}{|U|^2} = E_U(D|B)$$

$$+ \frac{|X'_{p_2} - Y'_{q_2}| - |X'_{p_1} - Y'_{q_1}|}{|U|^2} + \frac{|X'_{p_2}| - |X'_{p_2} \cap Y'_{q_2}|}{|U|^2} - \frac{|X'_{p_1}| - |X'_{p_1} \cap Y'_{q_1}|}{|U|^2} = E_U(D|B) + \frac{2|X'_{p_2} - Y'_{q_2}| - 2|X'_{p_1} - Y'_{q_1}|}{|U|^2}.$$

This completes the proof.

For convenience of introducing combination entropy, here gives a deformation of the definition of combination entropy (see Definition 1). According to $C_N^2 = (N(N-1))/2$, the following deformation can be got. Then the updating mechanism of combination conditional entropy is introduced on the basis of this deformation.

Definition 9. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. One can obtain the condition partition $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. The combination conditional entropy of B relative to D is defined as

$$CE(D|B) = \sum_{i=1}^{m} \left(\frac{|X_i|^2(|X_i| - 1)}{|U|^2(|U| - 1)} - \sum_{j=1}^{n} \frac{|X_i \cap Y_j|^2(|X_i \cap Y_j| - 1)}{|U|^2(|U| - 1)} \right). \tag{6}$$

Theorem 2. Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. The combination conditional entropy of D with respect to B is $E_U(D|B)$. Then, one can obtain the partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. $x \in X_{p_1}$ and $x \in Y_{q_1}$. If one and only object $x \in U$ is changed to x', then $x' \in X'_{p_2}$ and $x' \in Y'_{q_2}$ ($X'_{p_2} \in U_{x'}/B$ and $Y'_{q_2} \in U_{x'}/D$). The new combination conditional entropy becomes

$$CE_{U_{x'}}(D|B) = CE_U(D|B) + \Delta,$$

where $\Delta = \frac{|X'_{p_2} - Y'_{q_2}|(3|X'_{p_2}| + 3|X'_{p_2} \cap Y'_{q_2}| - 5) - |X'_{p_1} - Y'_{q_1}|(3|X'_{p_1}| + 3|X'_{p_1} \cap Y'_{q_1}| + 1)}{|U|^2}$, $X'_{p_1} = X_{p_1} - \{x\}$ and $Y'_{q_1} = Y_{q_1} - \{x\}$.

Proof. Similarly, when x is added to S, there are four same situations as the proof in Theorem 1. Then, the combination conditional entropy is

$$CE_{U_{x'}}(D|B) = \sum_{i=1}^{m-1} \left(\frac{|X_i|^2(|X_i| - 1)}{|U|^2(|U| - 1)} - \sum_{j=1}^{n-1} \frac{|X_i \cap Y_j|^2(|X_i \cap Y_j| - 1)}{|U|^2(|U| - 1)} - \frac{|X_i \cap Y'_{q_1}|^2(|X_i \cap Y'_{q_1}| - 1)}{|U|^2(|U| - 1)} - \frac{|X_i \cap Y'_{q_2}|^2(|X_i \cap Y'_{q_2}| - 1)}{|U|^2(|U| - 1)} \right)$$

$$+ \frac{|X'_{p_1}|^2(|X'_{p_1}| - 1)}{|U|^2(|U| - 1)} - \sum_{j=1}^{n-1} \frac{|X'_{p_1} \cap Y_j|^2(|X'_{p_1} \cap Y_j| - 1)}{|U|^2(|U| - 1)} - \frac{|X'_{p_1} \cap Y'_{q_1}|^2(|X'_{p_1} \cap Y'_{q_1}| - 1)}{|U|^2(|U| - 1)} - \frac{|X'_{p_1} \cap Y'_{q_2}|^2(|X'_{p_1} \cap Y'_{q_2}| - 1)}{|U|^2(|U| - 1)} + \frac{|X'_{p_2}|^2(|X'_{p_2}| - 1)}{|U|^2(|U| - 1)}$$

$$- \sum_{j=1}^{n-1} \frac{|X'_{p_2} \cap Y_j|^2(|X'_{p_2} \cap Y_j| - 1)}{|U|^2(|U| - 1)} - \frac{|X'_{p_2} \cap Y'_{q_1}|^2(|X'_{p_2} \cap Y'_{q_1}| - 1)}{|U|^2(|U| - 1)} - \frac{|X'_{p_2} \cap Y'_{q_2}|^2(|X'_{p_2} \cap Y'_{q_2}| - 1)}{|U|^2(|U| - 1)}$$

$$
\begin{aligned}
&= \sum_{i=1}^{m-1}\left(\frac{|X_i|^2(|X_i|-1)}{|U|^2(|U|-1)} - \sum_{j=1}^{n+1}\frac{|X_i \cap Y_j|^2(|X_i \cap Y_j|-1)}{|U|^2(|U|-1)}\right) + \frac{(|X_{p_1}|-1)^2(|X_{p_1}|-2)}{|U|^2(|U|-1)} - \sum_{j=1}^{n-1}\frac{|X_{p_1} \cap Y_j|^2(|X_{p_1} \cap Y_j|-1)}{|U|^2(|U|-1)}\\
&\quad - \frac{(|X_{p_1} \cap Y_{q_1}|-1)^2(|X_{p_1} \cap Y_{q_1}|-2)}{|U|^2(|U|-1)} - \frac{|X_{p_1} \cap Y_{q_2}|^2(|X_{p_1} \cap Y_{q_2}|-1)}{|U|^2(|U|-1)} + \frac{(|X_{p_2}|+1)^2|X_{p_2}|}{|U|^2(|U|-1)} - \sum_{j=1}^{n-1}\frac{|X_{p_2} \cap Y_j|^2(|X_{p_2} \cap Y_j|-1)}{|U|^2(|U|-1)}\\
&\quad - \frac{|X_{p_2} \cap Y_{q_1}|^2(|X_{p_2} \cap Y_{q_1}|-1)}{|U|^2(|U|-1)} - \frac{(|X_{p_2} \cap Y_{q_2}|+1)^2|X_{p_2} \cap Y_{q_2}|}{|U|^2(|U|-1)} = \sum_{i=1}^{m+1}\left(\frac{|X_i|^2(|X_i|-1)}{|U|^2(|U|-1)} - \sum_{j=1}^{n+1}\frac{|X_i \cap Y_j|^2(|X_i \cap Y_j|-1)}{|U|^2(|U|-1)}\right)\\
&\quad + \frac{-3|X_{p_1}|^2 + 5|X_{p_1}| + 3|X_{p_1} \cap Y_{q_1}|^2 - 5|X_{p_1} \cap Y_{q_1}|}{|U|^2(|U|-1)} + \frac{3|X_{p_2}|^2 + |X_{p_2}| - 3|X_{p_2} \cap Y_{q_2}|^2 - |X_{p_2} \cap Y_{q_2}|}{|U|^2(|U|-1)} = CE_U(D|B)\\
&\quad + \frac{|X_{p_2} - Y_{q_2}|(3|X_{p_2}| + 3|X_{p_2} \cap Y_{q_2}| + 1)}{|U|^2(|U|-1)} - \frac{|X_{p_1} - Y_{q_1}|(3|X_{p_1}| + 3|X_{p_1} \cap Y_{q_1}| - 5)}{|U|^2(|U|-1)}.
\end{aligned}
$$

And because $X'_{p_2} = X_{p_2} \cup \{x\}$ and $Y'_{q_2} = Y_{q_2} \cup \{x\}$, from Theorem 2, one can also get that

$$
CE_{U_{x'}}(D|B) = CE_U(D|B) + \frac{|X_{p_2} - Y_{q_2}|(3|X_{p_2}| + 3|X_{p_2} \cap Y_{q_2}| + 1)}{|U|^2(|U|-1)} - \frac{|X_{p_1} - Y_{q_1}|(3|X_{p_1}| + 3|X_{p_1} \cap Y_{q_1}| - 5)}{|U|^2(|U|-1)}.
$$

This completes the proof.

The following two theorems are the updating mechanisms of Shannon's conditional entropy shown in Definition 3.

Theorem 3. *Let $S = (U, C \cup D)$ be a decision table and $B \subseteq C$. The Shannon's conditional entropy of D with respect to B is $E_U(D|B)$. Then, one can obtain the partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. $x \in X_{p_1}$ and $x \in Y_{q_1}$. If one and only object $x \in U$ is changed to x', then $x' \in X'_{p_2}$ and $x' \in Y'_{q_2}$ ($X'_{p_2} \in U_{x'}/B$ and $Y'_{q_2} \in U_{x'}/D$). The new Shannon's conditional entropy becomes*

$$
H_{U_{x'}}(D|B) = H_U(D|B) - \Delta,
$$

where

$$
\begin{aligned}
\Delta &= \sum_{j=1, j \neq q_1}^{n}\frac{|X_{p_1} \cap Y_j|}{|U|}\log\frac{|X_{p_1}|}{|X'_{p_1}|} + \sum_{j=1, j \neq q_2}^{n}\frac{|X_{p_2} \cap Y_j|}{|U|}\log\frac{|X_{p_2}|}{|X'_{p_2}|} + \frac{|X_{p_1} \cap Y_{q_1}|}{|U|}\cdot\log\frac{|X'_{p_1} \cap Y'_{q_1}||X_{p_1}|}{|X_{p_1} \cap Y_{q_1}||X'_{p_1}|} + \frac{|X_{p_2} \cap Y_{q_2}|}{|U|}\log\frac{|X'_{p_2} \cap Y'_{q_2}||X_{p_2}|}{|X_{p_2} \cap Y_{q_2}||X'_{p_2}|}\\
&\quad + \frac{1}{|U|}\log\frac{|X'_{p_1}||X'_{p_2} \cap Y'_{q_2}|}{|X'_{p_2}||X'_{p_1} \cap Y'_{q_1}|}, X'_{p_1} = X_{p_1} - \{x\} \quad and \quad Y'_{q_1} = Y_{q_1} - \{x\}
\end{aligned}
$$
.

Proof. Similar to the proof in Theorem 4, it can be easily proved.

In view of that the formula of Δ is complicated, thus, for the large-scale data tables, an approximate computational formula is proposed in the following theorem.

Theorem 4. *Let $S = (U, C \cup D)$ be a large-scale decision table and $B \subseteq C$. The conditional Shannon's entropy of D with respect to B is $E_U(D|B)$. Then, one can obtain the partitions $U/B = \{X_1, X_2, \ldots, X_m\}$ and $U/D = \{Y_1, Y_2, \ldots, Y_n\}$. $x \in X_{p_1}$ and $x \in Y_{q_1}$. If one and only object $x \in U$ is changed to x', then $x' \in X'_{p_2}$ and $x' \in Y'_{q_2}$ ($X'_{p_2} \in U_{x'}/B$ and $Y'_{q_2} \in U_{x'}/D$). The new Shannon's conditional entropy becomes*

$$
H_{U_{x'}}(D|B) \approx H_U(D|B) - \frac{1}{|U|}\log\frac{|X'_{p_1}||X'_{p_2} \cap Y'_{q_2}|}{|X'_{p_2}||X'_{p_1} \cap Y'_{q_1}|},
$$

where $X'_{p_1} = X_{p_1} - \{x\}$ and $Y'_{q_1} = Y_{q_1} - \{x\}$.

Proof. Similarly, when x is added to S, there are four same situations as the proof in Theorem 1. Then, the Shannon's conditional entropy is

$$
\begin{aligned}
H_{U_{x'}}(D|B) &= -\left(\sum_{i=1}^{m-1}\frac{|X_i|}{|U|}\left(\sum_{j=1}^{n-1}\frac{|X_i \cap Y_j|}{|X_i|}\log\frac{|X_i \cap Y_j|}{|X_i|} + \frac{|X_i \cap Y'_{q_1}|}{|X_i|}\log\frac{|X_i \cap Y'_{q_1}|}{|X_i|} + \frac{|X_i \cap Y'_{q_2}|}{|X_i|}\log\frac{|X_i \cap Y'_{q_2}|}{|X_i|}\right)\right.\\
&\quad + \frac{|X'_{p_1}|}{|U|}\left(\sum_{j=1}^{n-1}\frac{|X'_{p_1} \cap Y_j|}{|X'_{p_1}|}\log\frac{|X'_{p_1} \cap Y_j|}{|X'_{p_1}|} + \frac{|X'_{p_1} \cap Y'_{q_1}|}{|X'_{p_1}|}\cdot\log\frac{|X'_{p_1} \cap Y'_{q_1}|}{|X'_{p_1}|} + \frac{|X'_{p_1} \cap Y'_{q_2}|}{|X'_{p_1}|}\cdot\log\frac{|X'_{p_1} \cap Y'_{q_2}|}{|X'_{p_1}|}\right)\\
&\quad \left.+ \frac{|X'_{p_2}|}{|U|}\left(\sum_{j=1}^{n-1}\frac{|X'_{p_2} \cap Y_j|}{|X'_{p_2}|}\log\frac{|X'_{p_2} \cap Y_j|}{|X'_{p_2}|} + \frac{|X'_{p_2} \cap Y'_{q_1}|}{|X'_{p_2}|}\log\frac{|X'_{p_2} \cap Y'_{q_1}|}{|X'_{p_2}|} + \frac{|X'_{p_2} \cap Y'_{q_2}|}{|X'_{p_2}|}\cdot\log\frac{|X'_{p_2} \cap Y'_{q_2}|}{|X'_{p_2}|}\right)\right)
\end{aligned}
$$

$$= -\left(\sum_{i=1}^{m-1}\frac{|X_i|}{|U|}(\sum_{j=1}^{n+1}\frac{|X_i\cap Y_j|}{|X_i|}log\frac{|X_i\cap Y_j|}{|X_i|}) + \frac{|X_{p_1}|}{|U|}\sum_{j=1}^{n-1}\frac{|X_{p_1}\cap Y_j|}{|X_{p_1}|}log\frac{|X_{p_1}\cap Y_j|}{|X'_{p_1}|} + \frac{|X_{p_1}|}{|U|}\cdot\frac{|X_{p_1}\cap Y_{q_1}|-1}{|X_{p_1}|}log\frac{|X'_{p_1}\cap Y'_{q_1}|}{|X'_{p_1}|} + \frac{|X_{p_1}|}{|U|}\cdot\right.$$

$$\frac{|X_{p_1}\cap Y_{q_2}|}{|X_{p_1}|}\cdot log\frac{|X'_{p_1}\cap Y'_{q_2}|}{|X'_{p_1}|} + \frac{|X_{p_2}|}{|U|}\sum_{j=1}^{n-1}\frac{|X_{p_2}\cap Y_j|}{|X_{p_2}|}\cdot log\frac{|X_{p_2}\cap Y_j|}{|X'_{p_2}|} + \frac{|X_{p_2}|}{|U|}\cdot\frac{|X_{p_2}\cap Y_{q_1}|}{|X_{p_2}|}log\frac{|X_{p_2}\cap Y_{q_1}|}{|X'_{p_2}|} + \frac{|X_{p_2}|}{|U|}\cdot\frac{|X_{p_2}\cap Y_{q_2}|+1}{|X_{p_2}|}\cdot$$

$$\left.log\frac{|X'_{p_2}\cap Y'_{q_2}|}{|X'_{p_2}|}\right) = -\left(\sum_{i=1}^{m-1}\frac{|X_i|}{|U|}\sum_{j=1}^{n+1}\frac{|X_i\cap Y_j|}{|X_i|}log\frac{|X_i\cap Y_j|}{|X_i|} + \frac{|X_{p_1}|}{|U|}\sum_{j=1}^{n-1}\frac{|X_{p_1}\cap Y_j|}{|X_{p_1}|}\left(log\frac{|X_{p_1}\cap Y_j|}{|X_{p_1}|} + log\frac{|X_{p_1}|}{|X'_{p_1}|}\right) + \frac{|X_{p_1}|}{|U|}\cdot\right.$$

$$\frac{|X_{p_1}\cap Y_{q_1}|}{|X_{p_1}|}\left(log\frac{|X_{p_1}\cap Y_{q_1}|}{|X_{p_1}|} + log\frac{|X'_{p_1}\cap Y'_{q_1}||X_{p_1}|}{|X_{p_1}\cap Y_{q_1}||X'_{p_1}|}\right) - \frac{1}{|U|}log\frac{|X'_{p_1}\cap Y'_{q_1}|}{|X'_{p_1}|} + \frac{|X_{p_1}|}{|U|}\cdot\frac{|X_{p_1}\cap Y_{q_2}|}{|X_{p_1}|}\cdot\left(log\frac{|X_{p_1}\cap Y_{q_2}|}{|X_{p_1}|} + log\frac{|X_{p_1}|}{|X'_{p_1}|}\right)$$

$$+ \frac{|X_{p_2}|}{|U|}\sum_{j=1}^{n-1}\frac{|X_{p_2}\cap Y_j|}{|X_{p_2}|}\cdot\left(log\frac{|X_{p_2}\cap Y_j|}{|X_{p_2}|} + log\frac{|X_{p_2}|}{|X'_{p_2}|}\right) + \frac{|X_{p_2}|}{|U|}\cdot\frac{|X_{p_2}\cap Y_{q_1}|}{|X_{p_2}|}\left(log\frac{|X_{p_2}\cap Y_{q_1}|}{|X_{p_2}|} + log\frac{|X_{p_2}|}{|X'_{p_2}|}\right) + \frac{|X_{p_2}|}{|U|}\cdot\frac{|X_{p_2}\cap Y_{q_2}|}{|X_{p_2}|}\cdot$$

$$\left.\left(log\frac{|X_{p_2}\cap Y_{q_2}|}{|X_{p_2}|} + log\frac{|X'_{p_2}\cap Y'_{q_2}||X_{p_2}|}{|X_{p_2}\cap Y_{q_2}||X'_{p_2}|}\right) + \frac{1}{|U|}log\frac{|X'_{p_2}\cap Y'_{q_2}|}{|X'_{p_2}|}\right).$$

To simplify the calculations of above formula, for the large-scale decision tables, here are some approximated expressions. In view of that $|X_{p_1}|$ and $|X_{p_2}|$ based on the large-scale decision tables are relatively large, respectively, one can get that $|X_{p_1}|\approx|X'_{p_1}|$ and $|X_{p_2}|\approx|X'_{p_2}|$ ($X'_{p_1}=X_{p_1}-\{x\}$ and $X'_{p_2}=X_{p_2}\cup\{x\}$), e.t. $log|X_{p_1}|/|X'_{p_1}|\approx 0$ and $log|X_{p_2}|/|X'_{p_2}|\approx 0$. Similarly, one can also get from $|X_{p_1}\cap Y_{q_1}|\approx|X'_{p_1}\cap Y'_{q_1}|$ and $|X_{p_2}\cap Y_{q_2}|\approx|X'_{p_2}\cap Y'_{q_2}|$ that $log|X'_{p_1}\cap Y'_{q_1}||X_{p_1}|/|X_{p_1}\cap Y_{q_1}||X'_{p_1}|\approx 0$ and $log|X'_{p_2}\cap Y'_{q_2}||X_{p_2}|/|X_{p_2}\cap Y_{q_2}||X'_{p_2}|\approx 0$. Hence, the above formula can be simplified to

$$H_{U_{x'}}(D|B)\approx -\left(\sum_{i=1}^{m}\frac{|X_i|}{|U|}\sum_{j=1}^{n}\frac{|X_i\cap Y_j|}{|X_i|}log\frac{|X_i\cap Y_j|}{|X_i|} - \frac{1}{|U|}log\frac{|X'_{p_1}\cap Y'_{q_1}|}{|X'_{p_1}|} + \frac{1}{|U|}\cdot log\frac{|X'_{p_2}\cap Y'_{q_2}|}{|X'_{p_2}|}\right) = H_U(D|B) - \frac{1}{|U|}log\frac{|X'_{p_2}\cap Y'_{q_2}||X'_{p_1}|}{|X'_{p_2}||X'_{p_1}\cap Y'_{q_1}|}.$$

Thus, this completes the proof.

5. Attribute reduction algorithm for decision tables with dynamically varying attribute values

Based on the updating mechanisms of the three entropies, this section introduces an attribute reduction algorithm based on information entropy for decision tables with dynamically varying attribute values. In view of that core is another key concept besides redcut in rough set theory [14,15], this section also gives an algorithm for core computation. In rough set theory, core is the intersection of all reducts of a given table, and core attributes are considered as the indispensable attributes in a reduct. Note that, for the three entropies, the following algorithms are commonly used to update core and reduct.

Algorithm 2. An algorithm to core computation for a dynamic decision table ($ACORE_{x'}$).

Input: A decision table $S=(U, C\cup D)$ and object $x\in U$ is changed to x'
Output: Core attribute $CORE_{x'}$ on $U_{x'}$
Step 1: Find X'_{p_1} and X'_{p_2}: in $U/C=\{X_1, X_2, \ldots, X_m\}$ and $x\in X_{p_1}$. If x is changed to x', and $x'\in X'_{p_2}$. One have $X'_{p_1}=X_{p_1}-\{x'\}$ and $U_{x'}/C = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X'_{p_2}, \ldots, X_m\}$.
Step 2: Find Y'_{q_1} and Y'_{q_2}: in $U/D=\{Y_1, Y_2, \ldots, Y_n\}$ and $x\in Y_{q_1}$. If x is changed to x', and $x'\in Y'_{q_2}$. We have $Y'_{q_1}=Y_{q_1}-\{x'\}$ and $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y'_{q_2}, \ldots, Y_n\}$.
Step 3: Compute $ME_{U_{x'}}(D|C)$ (according to Theorems 1, 2 or 4);
Step 4: $CORE_{U_{x'}}\leftarrow\emptyset$, for each $a\in C$
(1) In $U/(C-\{a\})=\{M_1, M_2, \ldots, M_{m'}\}$ ($m'\le m$) and $x\in M_{t_1}$. If x is changed to x', one have $x'\in M'_{t_2}$, $M'_{t_1}=M_{t_1}-\{x'\}$ and $U_{x'}/(C-\{a\}) = \{M_1, M_2, \ldots, M'_{t_1}, \ldots, M'_{t_2}, \ldots, M_{m'}\}$.
(2) Compute $ME_{U_{x'}}(D|C-\{a\})$ (according to Theorems 1, 2 or 4).
(3) If $ME_{U_{x'}}(D|C-\{a\})\ne ME_{U_{x'}}(D|C)$, then $CORE_{U_{x'}}=CORE_{U_{x'}}\cup\{a\}$.
Step 5: Return $CORE_{U_{x'}}$ and end.

Based on updating mechanisms of the three entropies, an attribute reduction algorithm for decision tables with dynamically varying attribute values is introduced in the following. In this algorithm, the existing reduction result is one of inputs, which is used to find its new reduct after data changes.

Algorithm 3. An algorithm to reduct computation for a dynamic decision table ($ARED_{x'}$).

Input: A decision table $S=(U, C\cup D)$, reduct RED_U on U, and the changed object x which is changed to x'
Output: Attribute reduct $RED_{U_{x'}}$ on $U_{x'}$

Table 1
Comparison of time complexity.

Entropy	Classic	Incremental
Core	$O(|U|^2)$	$O(|U||C| + max(|X'_{p_1}||Y'_{q_1}|, |X'_{p_2}||Y'_{q_2}|))$
	$CA\_CORE_{x'}$	$ACORE_{x'}$
Reduct	$O(|C||U|^2)$	$O(max(|C||X'_{p_1}||Y'_{q_1}|, |C||X'_{p_2}||Y'_{q_2}|))$
	$CA\_RED_{x'}$	$ARED_{x'}$
	$O(|C|^2|U| + |C||U|^2)$	$O(|C|^2|U| + max(|C||X'_{p_1}||Y'_{q_1}|, |C||X'_{p_2}||Y'_{q_2}|))$

Step 1: $B \leftarrow RED_U$. Find M'_{t_1} and M'_{t_2}: in $U/B = \{M_1, M_2, \ldots, M_{m'}\}$ and $x \in M_{t_1}$. If x is changed to x', and $x' \in M'_{t_2}$. One have $M'_{t_1} = M_{t_1} - \{x'\}$ and $U_{x'}/B = \{M_1, M_2, \ldots, M'_{t_1}, \ldots, M'_{t_2}, \ldots, M_{m'}\}$.

Step 2: Find Y'_{q_1} and Y'_{q_2}: in $U/D = \{Y_1, Y_2, \ldots, Y_n\}$ and $x \in Y_{q_1}$. If x is changed to x', and $x' \in Y'_{q_2}$. One have $Y'_{q_1} = Y_{q_1} - \{x'\}$ and $U_{x'}/D = \{Y_1, Y_2, \ldots, Y'_{q_1}, \ldots, Y'_{q_2}, \ldots, Y_n\}$.

Step 3: Compute $ME_{U_{x'}}(D|B)$ (according to Theorems 1, 2 or 4);

Step 4: Find X'_{p_1} and X'_{p_2}: in $U/C = \{X_1, X_2, \ldots, X_m\}$ and $x \in X_{p_1}$. If x is changed to x', and $x' \in X'_{p_2}$. One have $X'_{p_1} = X_{p_1} - \{x'\}$ and $U_{x'}/C = \{X_1, X_2, \ldots, X'_{p_1}, \ldots, X'_{p_2}, \ldots, X_m\}$.

Step 5: Compute $ME_{U_{x'}}(D|C)$ (according to Theorems 1, 2 or 4);

Step 6: If $ME_{U_{x'}}(D|B) = ME_{U_{x'}}(D|C)$, then $RED_{U_{x'}} \leftarrow RED_U$, turn to *Step 8*; else turn to *Step 7*.

Step 7: While $ME_{U_{x'}}(D|B) \neq ME_{U_{x'}}(D|C)$ *do*

{ For each $a \in C - B$, compute $Sig^{outer}_{U_{x'}}(a, B, D)$ (according to Theorems 1, 2 or 4 and Definition 6);

Select $a_0 = max\{Sig^{outer}_{U_{x'}}(a, B, D)\}$, $a \in C - B$;

$B \leftarrow B \cup \{a_0\}$.
}

Step 8: For each $a \in B$ *do*

{ Compute $Sig^{inner}_{U_{x'}}(a, B, D)$;

If $Sig^{inner}_{U_{x'}}(a, B, D) = 0$, then $B \leftarrow B - \{a\}$. }

Step 9: $RED_{U_{x'}} \leftarrow B$, return $RED_{U_{x'}}$ and end.

The following is time complexities of above two algorithms. First is the time complexity of computing entropy according to Theorems 1, 2, and 4, which is $O(m|C| + n + |X'_{p_1}||Y'_{q_1}| + |X'_{p_2}||Y'_{q_2}|) = O(max(|X'_{p_1}||Y'_{q_1}|, |X'_{p_2}||Y'_{q_2}|))$ (the explanations of m, n, X'_{p_1}, Y'_{q_1}, X'_{p_2} and Y'_{q_2} are shown in Theorems 1, 2, and 4). For convenience, Θ' is used to denote the above time complexity, i.e., $\Theta' = O(max(|X'_{p_1}||Y'_{q_1}|, |X'_{p_2}||Y'_{q_2}|))$.

In the algorithm $ACORE_{x'}$, the time complexity of steps 1–3 is Θ'; in step 4, the time complexity is $|C|\Theta'$. Hence, the time complexity of algorithm $ACORE_{x'}$ is

$$O(\Theta' + |C|\Theta') = O(|C|(max(|X'_{p_1}||Y'_{q_1}|, |X'_{p_2}||Y'_{q_2}|))) = O(max(|C||X'_{p_1}||Y'_{q_1}|, |C||X'_{p_2}||Y'_{q_2}|)).$$

In algorithm $ARED_{x'}$, the time complexity of steps 1–3 is Θ'; the time complexity of steps 4 and 5 is also $O(\Theta')$; in step 7, the time complexity of adding attributes is $O(|C|\Theta')$; in step 8, the time complexity of deleting redundant attributes is $O(|B|\Theta')$. Hence, the total time complexity of algorithm IA_RED_x is

$$O(\Theta' + |C|\Theta' + |B|\Theta') = O(|C|\Theta') = O(|C|^2|U| + max(|C||X'_{p_1}||Y'_{q_1}|, |C||X'_{p_2}||Y'_{q_2}|)).$$

To stress above findings, the time complexities of computing core and reduct are shown in Table 1. $CA\_CORE_{x'}$ and $CA\_RED_{x'}$ denote classic algorithms based on information entropy for computing core and reduct, respectively.

In Table 1, $|X'_{p_1}||Y'_{q_1}|$ (or $|X'_{p_2}||Y'_{q_2}|$) is usually much smaller than $|U|^2$. Hence, based on the three entropies, the calculation of proposed algorithms ($ACORE_{x'}$ and $ARED_{x'}$) are usually much smaller than that of the classic algorithms for reduct (or core).

6. Experimental analysis

The objective of the following experiments is to show effectiveness and efficiency of the proposed reduction algorithm $ARED_{x'}$. Due to that the core is a subset of a reduct, we only run the reduction algorithms in the experiments. Data sets used in the experiments are outlined in Table 2, which were all downloaded from UCI repository of machine learning databases. All the experiments have been carried out on a personal computer with Windows XP and Inter(R) Core(TM) 2 Quad CPU Q9400, 2.66 GHz and 3.37 GB memory. The software being used is Microsoft Visual Studio 2005 and programming language is C#.

Table 2
Description of data sets.

	Data sets	Samples	Attributes	Classes
1	Backup-large	307	35	19
2	Dermatology	366	33	6
3	Breast-cancer-wisconsin (Cancer)	683	9	2
4	Mushroom	5644	22	2
5	Letter-recognition (Letter)	20,000	16	26
6	Shuttle	58,000	9	7

Table 3
Comparison of reducts based on complementary entropy.

Data sets	CAR		$ARED_{x'}$	
	Reduct	Time/s	Reduct	Time/s
Backup-large	1,4,7,8,10,13,16,22	4.5937	1,4,7,8,10,13,22	0.1406
Dermatology	1,2,3,4,5,14,16,18,19	5.7812	1,2,3,4,5,14,18,19	0.2031
Cancer	1,2,4,6	2.0000	1,2,4,6	0.5000
Mushroom	1,2,3,5,7,8,9,20	486.2600	1,2,3,5,7,9,20	37.3120
Letter	1,2,3,4,5,8,9,10,11,12,13,15,16	9602.6000	1,2,3,4,5,8,9,10,11,12,15,16	358.81
Shuttle	1,2,3,5	17010.1000	1,2,3,5	5122.1000

There are two objectives to conduct the experiments. The first one is to show whether the reduct found by $ARED_{x'}$ is feasible by comparing with that of CAR (see in Section 6.1). The second one is to compare the efficiency of $ARED_{x'}$ and CAR (see in Section 6.2). Six UCI data sets are employed to test the two algorithms. *Mushroom* and *Breast-cancer-wisconsin* are data sets with missing values, and for a uniform treatment of all data sets, the objects with missing values have been removed. Moreover, *Shuttle* is preprocessed using the data tool Rosetta.

6.1. Effectiveness analysis

In this subsection, to test the effectiveness of $ARED_{x'}$, four common evaluation measures in rough set theory are employed to evaluate the decision performance of the reducts found by CAR and $ARED_{x'}$. The four evaluation measures are approximate classified precision, approximate classified quality, certainty measure and consistency measure.

In [14], Pawlak defined the approximate classified precision and approximate classified quality to describe the precision of approximate classification in rough set theory.

Definition 10. Let $S = (U, C \cup D)$ be a decision table and $U/D = \{X_1, X_2, \ldots, X_r\}$. The approximate classified precision of C with respect to D is defined as

$$AP_C(D) = \frac{|POS_C(D)|}{\sum_{i=1}^{r} |\overline{C}X_i|}. \tag{7}$$

Definition 11. Let $S = (U, C \cup D)$ be a decision table. The approximate classified quality of C with respect to D is defined as

$$AQ_C(D) = \frac{|POS_C(D)|}{|U|}. \tag{8}$$

In rough set theory, by adopting reduction algorithms, one can get reducts for a given decision table. Then, based on a reduct, a set of decision rules can be generated from a decision table. Here briefly recalls the notions of decision rules [14,52], which will be used in the following development.

Definition 12. Let $S = (U, C \cup D)$ be a decision table. $U/C = \{X_1, X_2, \ldots, X_m\}$, $U/D = \{Y_1, Y_2, \ldots, Y_n\}$ and $\cap Y_j \neq \emptyset$. $des(X_i)$ and $des(Y_j)$ are denoted the descriptions of the equivalence classes X_i and Y_j, respectively. A decision rule induced by C is formally defined as

$$Z_{ij} : des(X_i) \rightarrow des(Y_j), X_i \in U/C, Y_j \in U/D. \tag{9}$$

In [53,54], certainty measure and support measure were introduced to evaluate a single decision rule. For a rule set, two measures were introduced to measure the certainty and consistency in [15]. However, it has been pointed out that those two measures cannot give elaborate depictions of the certainty and consistency for a rule set in [52]. To address this issue, Qian et al. [52] defined certainty measure and consistency measure to evaluate the certainty and consistency of a set of decision rules, which has attracted considerable attention [55].

Definition 13. Let $S = (U, C \cup D)$ be a decision table, $U/C = \{X_1, X_2, \ldots, X_m\}$, $U/D = \{Y_1, Y_2, \ldots, Y_n\}$, and $RULE = \{Z_{ij} | Z_{ij} : des(X_i) \rightarrow des(Y_j), X_i \in U/C, Y_j \in U/D\}$. The certainty measure α of the decision rules on S is defined as

$$\alpha(S) = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{|X_i \cap Y_j|^2}{|U||X_i|}. \tag{10}$$

Definition 14. Let $S = (U, C \cup D)$ be a decision table, $U/C = \{X_1, X_2, \ldots, X_m\}$, $U/D = \{Y_1, Y_2, \ldots, Y_n\}$, and $RULE = \{Z_{ij} | Z_{ij} : des(X_i) \rightarrow des(Y_j), X_i \in U/C, Y_j \in U/D\}$. The consistency measure β of the decision rules on S is defined as

$$\beta(S) = \sum_{i=1}^{m} \frac{|X_i|}{|U|} \left[1 - \frac{4}{|X_i|} \sum_{j=1}^{n} \frac{|X_i \cap Y_j|^2}{|X_i|} \left(1 - \frac{|X_i \cap Y_j|}{|X_i|} \right) \right]. \tag{11}$$

For each data set in Table 2, 50% objects are selected randomly and replaced by new ones. Then, algorithms CAR and $ARED_{x'}$ are employed to update reduct of each varying data set. The generated reducts are shown in Tables 3, 5 and 7, and the evaluation results of reducts based on the four evaluation measures are shown in Tables 4, 6 and 8.

- Comparison of CAR and $ARED_{x'}$ based on complementary entropy

It is easy to note from Table 4 the values of the four evaluation measures of the generated reducts by using the two algorithms are very close, and even identical on some data sets. But, according to Table 3, the computational time of $ARED_{x'}$ is much smaller than that of CAR. In other words, the performance and decision making of the reduct found by $ARED_{x'}$ are very close to that of CAR, but $ARED_{x'}$ is more efficient.

Table 4
Comparison of evaluation measures based on complementary entropy.

Data sets	CAR				$ARED_{x'}$			
	AQ	AP	α	β	AQ	AP	α	β
Backup-large	1.0000	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000
Dermatology	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Cancer	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Mushroom	1.0000	1.0000	1.0000	1.0000	0.9996	0.9993	0.9993	0.9986
Letter	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Shuttle	0.9988	0.9977	0.9976	0.9953	0.9988	0.9977	0.9976	0.9953

Table 5
Comparison of reducts based on combination entropy.

Data sets	CAR		$ARED_{x'}$	
	Reduct	Time/s	Reduct	Time/s
Backup-large	1,4,7,8,10,13,16,22	4.5312	1,4,7,8,10,13,22	0.1366
Dermatology	1,2,3,4,5,14,16,18,19	5.7500	1,2,3,4,5,14,18,19	0.2030
Cancer	1,2,4,6	1.9843	1,2,4,6	0.4987
Mushroom	1,2,3,4,7,8,9,20	478.3700	1,2,3,4,7,8,9,20	37.3010
Letter	1,2,3,4,5,8,9,10,11,12,13,15,16	8825.6000	1,2,3,4,5,8,9,11,12,13,15,16	358.8100
Shuttle	1,2,3,5	19935.7000	1,2,3,5	5089.1000

Table 6
Comparison of evaluation measures based on combination entropy.

Data sets	CAR				$ARED_{x'}$			
	AQ	AP	α	β	AQ	AP	α	β
Backup-large	1.0000	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000
Dermatology	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Cancer	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Mushroom	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Letter	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Shuttle	0.9988	0.9977	0.9976	0.9953	0.9988	0.9977	0.9976	0.9953

Table 7
Comparison of reducts based on Shannon's entropy.

Data sets	CAR		$ARED_{x'}$	
	Reduct	Time/s	Reduct	Time/s
Backup-large	1,4,5,6,7,8,10,16,22	5.3281	1,4,5,6,7,8,10,22	0.1368
Dermatology	1,4,5,9,12,14,17,18,21,22,26	5.7500	1,4,5,9,12,14,17,18,26	0.2030
Cancer	2,3,5,6	1.9843	2,3,5,6	0.5125
Mushroom	1,2,3,4,5,9,20,22	482.75	1,2,3,5,9,20,22	37.7510
Letter	1,2,3,4,5,8,9,10,11,12,13,15,16	8389.8000	1,2,3,4,5,8,9,10,11,13,16	358.8700
Shuttle	1,2,3,5	23698.5000	1,2,3,5	5130.6000

Table 8
Comparison of evaluation measures based on Shannon's entropy.

Data sets	CAR				$ARED_{x'}$			
	AQ	AP	α	β	AQ	AP	α	β
Backup-large	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Dermatology	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Cancer	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Mushroom	1.0000	1.0000	1.0000	1.0000	0.9996	0.9993	0.9993	0.9986
Letter	0.9999	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000
Shuttle	0.9988	0.9977	0.9976	0.9953	0.9988	0.9977	0.9976	0.9953

Hence, the experimental results indicate that, compared with the classic reduction algorithm *CAR* based on complementary entropy, the algorithm $ARED_{x'}$ can find a feasible reduct in a much shorter time.

- Comparison of *CAR* and $ARED_{x'}$ based on combination entropy

From Tables 5 and 6, it is easy to get that algorithm $ARED_{x'}$ can find a reduct which has same performance and decision making as those of reduct generated by *CAR* in a much shorter time. Thus, compared with *CAR* based on combination entropy, the algorithm $ARED_{x'}$ is more efficient to deal with dynamic data sets.

- Comparison of *CAR* and $ARED_{x'}$ based on Shannon's entropy

Table 9
Comparison of computational time based on complementary entropy.

Data sets	CAR					$ARED_{x'}$				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Backup-large	4.6875	4.8594	5.0625	4.4688	4.5937	0.0312	0.0468	0.0781	0.1250	0.1406
Dermatology	5.3906	5.5781	5.6093	6.6250	5.7812	0.0468	0.0781	0.1250	0.1562	0.2031
Cancer	1.4843	1.5937	1.7187	1.9062	2.0000	0.0781	0.1718	0.2656	0.4062	0.5000
Mushroom	221.0100	283.7800	368.7100	411.2800	468.2600	4.3125	9.9218	17.9060	29.0000	37.3120
Letter	8133.2000	8283.2000	8630.1000	9260.2000	9602.6000	83.6250	140.2500	220.4800	292.6800	358.8100
Shuttle	12909.8000	13905.3000	14523.9000	16100.1000	17010.1000	776.1000	1707.3000	2979.3000	4098.7000	5122.1000

Table 10
Comparison of computational time based on combination entropy.

Data sets	CAR					$ARED_{x'}$				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Backup-large	4.8125	4.8750	5.0468	4.4531	4.5312	0.0412	0.0568	0.0781	0.1193	0.1366
Dermatology	5.3906	5.4687	5.5156	5.7500	5.7500	0.0478	0.0781	0.1250	0.1563	0.2030
Cancer	1.4843	1.5781	1.7187	1.9062	1.9843	0.0781	0.1718	0.2656	0.4063	0.4987
Mushroom	222.7500	253.1500	371.0600	424.5100	478.3700	4.3712	9.9118	17.8600	28.9370	37.3010
Letter	7201.4000	7669.8000	8011.8000	8485.2000	8825.6000	83.5050	142.0500	200.4500	289.6600	358.8100
Shuttle	14122.1000	15468.9000	16236.1000	18896.9000	19935.7000	758.1100	1668.3000	3005.3000	4120.8000	5089.1000

Table 11
Comparison of computational time based on Shannon's entropy.

Data sets	CAR					$ARED_{x'}$				
	10%	20%	30%	40%	50%	10%	20%	30%	40%	50%
Backup-large	4.5937	4.7343	5.0000	5.2187	5.3281	0.0402	0.0568	0.0781	0.1194	0.1368
Dermatology	5.1875	5.5781	4.2500	6.3125	6.6562	0.0478	0.0750	0.1250	0.1565	0.2036
Cancer	1.4843	1.6093	1.7500	1.8750	1.9843	0.0781	0.1709	0.2625	0.4063	0.5125
Mushroom	221.8400	256.3600	330.3100	370.5900	482.7500	4.3825	9.9288	17.8760	28.9370	37.7510
Letter	7189.7000	7502.8000	7686.7000	8015.2000	8389.8000	83.6250	141.2500	223.4800	295.4800	358.8700
Shuttle	12968.3000	15126.7000	17356.1000	20123.3000	23698.5000	775.1000	1705.3000	2985.3000	4101.5000	5130.6000

According to experimental results in Tables 7 and 8, it is easy to see that performance and decision making of reducts generated by $ARED_{x'}$ and CAR are relatively close. But, $ARED_{x'}$ is more efficient than CAR. Hence, one can observe that algorithm $ARED_{x'}$ can find a feasible reduct, and save lots of computational time.

6.2. Efficiency analysis

The objective of experiments in this subsection is to further illustrate efficiency of algorithm $ARED_{x'}$. For each data set in Table 2, 10%, 20%, . . ., 50% objects are selected, in order, and are replaced by new ones. For each data set after each variation (from 10% to 50%), algorithms CAR and $ARED_{x'}$ are used to update reducts, respectively. The efficiency of the two algorithms are demonstrated by comparing their computational time. Experimental results are shown in Tables 9–11. 10%, 20%, . . ., 50% in the tables mean 10%, 20%, . . ., 50% objects with data values being varied, respectively.

Based on the three entropies, it is easy to see from Tables 9–11 that, for each data set after each variation, the computational time of algorithm $ARED_{x'}$ is much smaller than that of the classic reduction algorithm CAR, especially for the larger data sets Mushroom and Letter. In addition, with the number of varying objects increasing (from 10% to 50%), the computational time of $ARED_{x'}$ is always much smaller than that of CAR. Hence, the experimental results show that algorithm $ARED_{x'}$ is efficient to solving data sets with dynamically varying data values.

6.3. Related discussion

This subsection summarizes the advantages of algorithm $ARED_{x'}$ for generating reduct and offers explanatory comments. Obviously, in above two subsections, the experimental results better illustrate effectiveness and efficiency of $ARED_{x'}$.

- Algorithm $ARED_{x'}$ based on each of the three entropies can find a feasible reduct of a given dynamic decision table.
  According to experimental results in Section 6.1, it is easy to get that the decision performance of reducts generated by CAR and $ARED_{x'}$ are very close, and even identical on some data sets. Hence, compared with the classic reduction algorithms based on the three entropies, the reduct generated by $ARED_{x'}$ can be considered as a feasible reduct.
- Compared with the classic reduction algorithms (CAR) based the three entropies, $ARED_{x'}$ finds a reduct in a very efficient manner.
  Experimental results in Section 6.2 show that, based on the three entropies, the computational time of generating reduct by using $ARED_{x'}$ is much shorter than that of CAR.
- The development in the paper may make an important contribution to deal with large-scale dynamic data sets in applications.

The experimental results show that the efficiency of $ARED_{x'}$ is obvious in solving large-scale dynamic data sets. In reality, acquiring knowledge from large-scale complicated data sets is still a challenging issue. It is our wish that this paper provides new techniques for dealing with large-scale dynamic data sets.

7. Conclusions and future work

Feature selection for dynamic data sets is still a challenging issue in the field of artificial intelligence. In this paper, based on three representative entropies, an attribute reduction algorithm is proposed to update reduct of data sets with dynamically varying data values. The experimental results show that, compared with the classic reduction algorithms based on the three entropies, this algorithm can generate a feasible reduct in a much shorter time. It is our wish that this study provides new views and thoughts on dealing with large-scale and complicated dynamic data sets in applications.

It should be pointed out that updating mechanisms of the three entropies introduced in this paper are only applicable when data are varied one by one, whereas many real data may vary in groups in application. This gives rise to many difficulties for the proposed feature selection algorithm to deal with. Hence, it is expected to carry out the following work to improve efficiency of selecting useful features in dynamic data sets in the future:

- Developing group updating mechanisms of entropies and relative feature selection algorithms.
- Discernibility matrix is one of key concepts in rough set. Future work may include analyzing discernibility matrix for data sets with dynamically varying data values.
- Designing efficient feature selection algorithms based on generalized rough set models such as incomplete rough set model, dominance rough set model and multi-granulation rough set model.

References

[1] F. Hu, G.Y. Wang, H. Huang, Y. Wu, Incremental attribute reduction based on elementary sets, in: Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Regina, Canada, 2005, pp. 185–193.
[2] J.Y. Liang, W. Wei, Y.H. Qian, An incremental approach to computation of a core based on conditional entropy, Chinese Journal of System Engineering Theory and Practice 4 (2008) 81–89.
[3] D. Liu, T.R. Li, D. Ruan, W.L. Zou, An incremental approach for inducing knowledge from dynamic information systems, Fundamenta Informaticae 94 (2009) 245–260.
[4] M. Orlowska, Maintenance of knowledge in dynamic information systems, in: R. Slowinski (Ed.), Proceeding of the Intelligent Decision Support, Handbook of Applications and Advances of the Rough Set Theory, Kluwer Academic Publishers, Dordrecht, 1992, pp. 315–330.
[5] L. Shan, W. Ziarko, Data-based acquisition and incremental modification of classification rules, Computational Intelligence 11 (1995) 357–370.
[6] M. Yang, An incremental updating algorithm for attributes reduction based on the improved discernibility matrix, Chinese Journal of Computers 30 (5) (2007) 815–822.
[7] Z. Zheng, G. Wang G, RRIA: a rough set and rule tree based incremental knowledge acquisition algorithm, Fundamenta Informaticae 59 (2004) 299–313.
[8] C.C. Chan, A rough set approach to attribute generalization in data mining, Information Science 107 (1998) 169–176.
[9] T.R. Li, D. Ruan, W. Geert, J. Song, Y. Xu, A rough sets based characteristic relation approach for dynamic attribute generalization in data mining, Knowledge-Based System 20 (2007) 485–494.
[10] Y. Cheng, The incremental method for fast computing the rough fuzzy approximations, Data & Knowledge Engineering 70 (2011) 84–100.
[11] D. Liu, J.B. Zhang, T.R. Li, A probabilistic rough set approach for incremental learning knowledge on the change of attribute, in: Proceedings 2010 International Conference on Foundations and Applications of Computational Intelligence, 2010, pp. 722–727.
[12] H.M. Chen, T.R. Li, S.J. Qiao, D. Ruan, A rough set based dynamic maintenance approach for approximations in coarsening and refining attribute values, International Journal of Intelligent Systems 25 (2010) 1005–1026.
[13] D. Liu, T.R. Li, G.R. Liu, P. Hu, An incremental approach for inducing interesting knowledge based on the change of attribute values, in: Proceedings 2009 IEEE International Conference on Granular Computing, Nanchang, China, 2009, pp. 415–418.
[14] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Boston, 1991.
[15] Z. Pawlak, A. Skowron, Rudiments of rough sets, Information Sciences 177 (1) (2007) 3–27.
[16] H.S. Own, A. Abraham, A new weighted rough set framework based classification for Egyptian NeoNatal Jaundice, Applied Soft Computing 12 (3) (2012) 999–1005.
[17] K. Kaneiwa, A rough set approach to multiple dataset analysis, Applied Soft Computing 11 (2) (2011) 2538–2547.
[18] A.K. Das, J. Sil, An efficient classifier design integrating rough set and set oriented database operations, Applied Soft Computing 11 (2) (2011) 2279–2285.
[19] P. Dey, S. Dey, S. Datta, J. Sil, Dynamic discreduction using rough sets, Applied Soft Computing 11 (5) (2011) 3887–3897.
[20] R.R. Chen, Y.I. Chiang, P.P. Chong, Y.H. Lin, H.K. Chang, Rough set analysis on call center metrics, Applied Soft Computing 11 (4) (2011) 3804–3811.
[21] W. Wei, J.Y. Liang, Y.H. Qian, A comparative study of rough sets for hybrid data, Information Sciences 190 (1) (2012) 1–16.
[22] D.G. Chen, Q.H. Hu, Y.P. Yang, Parameterized attribute reduction with Gaussian kernel based fuzzy rough sets, Information Sciences 181 (23) (2011) 5169–5179.
[23] Q.H. Hu, Z.X. Xie, D.R. Yu, Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation, Pattern Recognition 40 (2007) 3509–3521.
[24] Q.H. Hu, D.R. Yu, Z.X. Xie, Information-preserving hybrid data reduction based on fuzzy-rough techniques, Pattern Recognition Letters 27 (5) (2006) 414–423.
[25] Y.Y. Yao, The superiority of three-way decisions in probabilistic rough set models, Information Sciences 181 (6) (2011) 1080–1096.
[26] G. Gediga, I. Düntsch, Rough approximation quality revisited, Artificial Intelligence 132 (2001) 219–234.
[27] R. Jensen, Q. Shen, Fuzzy-rough sets assisted attribute selection, IEEE Transactions on Fuzzy Systems 15 (1) (2007) 73–89.
[28] J.Y. Liang, Y.H. Qian, Information granules and entropy theory in information systems, Science in China (Series F) 51 (10) (2008) 1427–1444.
[29] J.S. Mi, Y. Leung, W.Z. Wu, An uncertainty measure in partition-based fuzzy rough sets, International Journal General Systems 34 (2005) 77–90.
[30] W. Pedrycz, G. Vukovich, Feature analysis through information granulation and fuzzy sets, Pattern Recognition 35 (2002) 825–834.
[31] Y.H. Qian, C.Y. Dang, J.Y. Liang, D.W. Tang, Set-valued ordered information systems, Information Sciences 179 (2009) 2809–2832.
[32] Z.B. Xu, J.Y. Liang, C.Y. Dang, K.S. Chin, Inclusion degree: a perspective on measures for rough set data analysis, Information Sciences 141 (2002) 227–236.
[33] J.Y. Liang, K.S. Chin, C.Y. Dang, C.M. Yam Richid, A new method for measuring uncertainty and fuzziness in rough set theory, International Journal of General Systems 31 (4) (2002) 331–342.
[34] J.Y. Liang, Z.Z. Shi, The information entropy, rough entropy and knowledge granulation in rough set theory, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 12 (1) (2004) 37–46.
[35] Y.H. Qian, J.Y. Liang, Combination entropy and combination granulation in rough set theory, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 16 (2) (2008) 179–193.
[36] C.E. Shannon, The mathematical theory of communication, The Bell System Technical Journal 27 (3,4) (1948) 373–423.
[37] Y.H. Qian, J.Y. Liang, W. Pedrycz, C.Y. Dang, Positive approximation: an accelerator for attribute reduction in rough set theory, Artificial Intelligence 174 (2010) 597–618.
[38] G.Y. Wang, H. Yu, D.C. Yang, Decision table reduction based on conditional information entropy, Chinese Journal of Computer 25 (7) (2002) 759–766.

[39] M. Dash, H. Liu, Consistency-based search in feature selection, Artificial Intelligence 151 (2003) 155–176.
[40] A. Skowron, 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 Academic Publisher, Dordrecht, 1992.
[41] M. Kryszkiewicz, P. Lasek, FUN: fast discovery of minimal sets of attributes functionally determining a decision attribute, Transactions on Rough Sets 9 (2008) 76–95.
[42] W. Ziarko, Variable precision rough set model, Journal of Computer and System Science 46 (1) (1993) 39–59.
[43] Y.Y. Yao, Decision-theoretic rough set models, Lecture Notes in Artificial Intelligence 4481 (2007) 1–12.
[44] Y.Y. Yao, Y. Zhao, Attribute reduction in decisiontheoretic rough set models, Information Sciences 178 (2008) 3356–3373.
[45] S. Greco, B. Matarazzo, R. Slowinski, Rough approximation by dominance relations, International Journal of Intelligent Systems 17 (2002) 153–171.
[46] Y.H. Qian, C.Y. Dang, J.Y. Liang, D.W. Tang, Set-valued ordered information systems, Information Sciences 179 (2009) 2809–2832.
[47] M.W. Shao, W.X. Zhang, Dominance relation and rules in an incomplete ordered information system, International Journal of Intelligent Systems 20 (2005) 13–27.
[48] D. Dubois, H. Prade, Rough fuzzy sets and fuuzy rough sets, International Journal of General Systems 17 (1990) 191–209.
[49] X.H. Hu, N. Cercone, Learning in relational databases: a rough set approach, International Journal of Computational Intelligence 11 (2) (1995) 323–338.
[50] D. Slezak, Approximate entropy reducts, Fundamenta Informaticae 53 (3-4) (2002) 365–390.
[51] Z.Y. Xu, Z.P. Liu, B.R. Yang, W. Song, A quick attribute reduction algorithm with complexity of max($O(|C||U|)$, $O(|C|^2|U/C|)$), Chinese Journal of Computer 29 (3) (2006) 391–398.
[52] Y.H. Qian, J.Y. Liang, D.Y. Li, H.Y. Zhang, C.Y. Dang, Measures for evaluating the decision performance of a decision table in rough set theory, Information Sciences 178 (2008) 181–202.
[53] I. Düntsch, G. Gediga, Uncertainty measures of rough set prediction, Artificial Intelligence 106 (1998) 109–137.
[54] V.N. Huynh, Y. Nakamori, A roughness measure for fuzzy sets, Information Sciences 173 (2005) 255–275.
[55] W. Wei, J.Y. Liang, Y.H. Qian, F. Wang, C.Y. Dang, Comparative study of decision performance of decision tables induced by attribute reductions, International Journal of General Systems 39 (8) (2010) 813–838.