BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **16**, No 2
Sofia • 2016 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2016-0016

Metric Based Attribute Reduction Method in Dynamic Decision Tables

Demetrovics Janos[1], *Nguyen Thi Lan Huong*[2], *Vu Duc Thi*[3], *Nguyen Long Giang*[4]

[1]*Institute for Computer and Control* (*SZTAKI*), *Hungarian Academy of Sciences, Hungary*
[2]*People's Police Academy, Viet Nam*
[3]*Institute of Information Technology, VNU, Viet Nam*
[4]*Institute of Information Technology, VAST, Viet Nam*
Emails: demetrovics@sztaki.mta.hu lanhuongnguyenppa@gmail.com vdthi@vnu.edu.vn nlgiang@ioit.ac.vn

Abstract: *Feature selection is a vital problem which needs to be effectively solved in knowledge discovery in databases and pattern recognition due to two basic reasons: minimizing costs and accurately classifying data. Feature selection using rough set theory is also called attribute reduction. It has attracted a lot of attention from researchers and numerous potential results have been gained. However, most of them are applied on static data and attribute reduction in dynamic databases is still in its early stages. This paper focuses on developing incremental methods and algorithms to derive reducts, employing a distance measure when decision systems vary in condition attribute set. We also conduct experiments on UCI data sets and the experimental results show that the proposed algorithms are better in terms of time consumption and reducts' cardinality in comparison with non-incremental heuristic algorithm and the incremental approach using information entropy proposed by authors in* [17].

Keywords: *Rough set, decision systems, attribute reduction, reduct, metric*.

1. Introduction

Feature selection plays a critical role in data mining and machine learning. It arises from the need to improve time consumption, the overburden when dealing with analysis large data and the abundance of data. Feature selection aims to decide on taking a subset consisting of most relevant attributes which preserve the interpretation as the original attribute set and to validate the subset with respect to the analysis goals.

Rough set theory by Professor Zdzislaw Pawlak [13] right after its start-up has been recognized as a mathematical tool for data analysis, especially vague data. The philosophy of rough set theory is to approximate unknown concepts by existing knowledge structure in a knowledge base. Rough set-based feature selection means to determine an appropriate subset of attributes that can preserve the ability to classify the set of objects without transformation the data.

In the latest fifteen years, there is a rapid growth of interests to attribute reduction. Among proposed techniques, heuristic methods are the most popular. The common points of heuristic methods typically consist of two steps: the first one is to define evaluation criteria for chosen features and the second is to give strategies for searching [1, 2, 5-8, 12, 16, 17]. J. Liang and other authors (see [9, 18]) classify attribute reduction methods based on heuristic functions into four categories: positive-region reduction, Shannon's entropy reduction, Liang's entropy reduction and combination entropy reduction.

However, the amount of research on dynamic databases is inadequate. Meanwhile, in the real world today, data usually is updated and changed with time. It can vary from adding or deleting objects or features to updating existing objects. In these cases a straightforward solution could be applied, that is to carry repeatedly the exist algorithms to finding new reduced feature subsets. Nevertheless, the time spent in re-computation is not trivial, especially when the number of objects or features being added, deleted or updated is large. This triggers research on new methods for dynamic information systems. Among them, the most effective techniques usually are incremental methods.

Recent studies on incremental methods to find reducts on dynamic decision tables have employed different evaluation criteria. In [7, 8, 20], authors used positive region and discernibility matrix for reduction algorithms when adding new objects. W. Qian and other authors proposed an incremental algorithm for feature reduction in decision tables using dependency function in the case of adding or deleting a feature subset (see [16]). Using information entropy is a trend which has been mentioned in a number of studies. In [11, 17], authors constructed formulas for updating three entropies (Shannon's entropy, Liang entropy and combination entropy) when adding or deleting objects. Using these entropies, they propose incremental algorithms for finding reducts. In [5], authors deal with the variation of decision tables from the angle of updating one object with a similar philosophy as methods in [11, 17]. However, the criteria proposed are quite complex in terms of representative formulas and have not yet completely dealt with dynamic decision tables. To deal with the variation of the decision tables and complex formulas, we propose using metric distance - a major effective approach to understand the different between two objects in algebra, geometry, set theory… Liang, Li and Qian [9] review the measures in rough set theory from the angle of distance. In this study, authors prove that most measures in rough set theory can be converted to distance measures. So that, using distance metric could be a potential trend for the simplicity in representation and computation but remain effective on rough set-based results.

Up to date, according to our best knowledge, not many researches on distance-based feature selection have been conducted. In [2], authors use variant Jaccard distance for the reduct algorithm and prove that their methods belong to the group of Liang entropy methods mentioning above. However, this research is not completely different from entropy methods – in [2] authors note that the distance used in proposed algorithms actually is equal to the conditional Liang entropy [10]. In [12], authors construct a metric distance and propose the metric-based algorithm for attribute reduction. They also prove that the reducts are better to compare with Liang's entropy methods in terms of cardinality of reducts. However, the method can only be applied for static decision tables. The potential of using metric in attribute reduction and the incompleteness of the work in [12] motivates us to develop the study. In this paper, we use the proposed distance to construct incremental algorithms for finding reducts in dynamic decision systems in cases of adding and deleting a subset of condition attributes. To corroborate advantages of the algorithms, experiments on UCI data sets are conducted to compare our method with two other cases: Re-implementing the reduct heuristic algorithm for the varied attribute set and the incremental method using entropies in [17].

The paper is organized as follows. Basic concepts of rough set theory and metric based reduction [12] are recalled in Section 2. In Section 3, at first we work out the incremental formulas to infer the metric from the original one after a subset of condition attribute is added or deleted from decision systems. Then we suggest two algorithms for finding reducts. Experimental analysis is presented in Section 4. Section 5 is the conclusion of the paper and further points for research in the future.

2. Preliminaries

In this section we summary some basic concepts in rough set theory [13] and metric based attribute reduction [12].

Definition 1 (information system). An information system is a couple IS = (U, A) where U is a finite nonempty set of objects and A is a finite nonempty set of features. Each $a \in A$ determines a map $a: U \to V_a$ where V_a is the value set of a.

Definition 2 (indiscernibility relation). Given information system IS = (U, A), for each $P \subseteq A$ a binary relation IND(P) on U which is also called indiscernibility relation is defined as $\mathrm{IND}(P) = \{(u, v) \in U \times U | \forall a \in P, a(u) = a(v)\}$.

IND(P) is an equivalence relation and determines a partition of U. Let U/IND(P) (briefly U/P) denotes a family of all equivalence classes of the relation IND(P) and $[u]_P = \{v \in U \mid (u, v) \in \mathrm{IND}(P)\ \forall u \in U\}$. Then, $U/P = \{[u]_P \mid u \in U\}$.

Given an information system IS = (U, A), $B \subseteq A$ and $X \subseteq U$, $\underline{B}X = \{u \in U | [u]_B \subseteq X$ and $\overline{B}X = \{u \in U | [u]_B \cap X \neq \emptyset\}$ are respectively called B-lower approximation and B-upper approximation of X respect to B. These two sets are used to approximate the set X in rough set theory.

A decision system (or decision table) is an information system (U, A), where A includes two separate subsets: condition attribute subset C and decision attribute subset D. So that, a Decision System (DS) could be written as DS $= (U, C \cup D)$ where $C \cap D = \emptyset$.

Assuming that $\mathrm{DS} = (U, C \cup D)$ is a decision system, D_i denotes an equivalence class of partition U/D, then $U/D = \{D_i\}$. $\mathrm{POS}_C(D) = \bigcup_{D_i \in U/D} (\underline{C}D_i)$ is called C-positive region of D. One can easily obtain that $\mathrm{POS}_C(D)$ is a set of objects belonging to U which are partitioned by IND(C) right into decision classes of U/D.

In an information system, not all the features are necessary for recognition or classification [3]. The concept of attribute reduction was first introduced by Pawlak [13] which targets to eliminate irrelevant or redundant features in such that the discernible ability of the attribute set is preserved. The remaining feature set after elimination is called a reduct [3, 4].

Research directions related to reduct in decision tables have attracted the interest of many researchers in recent years [14, 15] in which attribute reduction is one of important topics. Attribute reduction methods generally consists of two basic steps: Feature evaluation and searching mechanism. In the first step, up to now many proposals have been suggested while in the second step, most of searching strategies are heuristic. In the next, we briefly present the attribute reduction method using a metric as criteria for attribute evaluation in [12].

Definition 3 [12]. Given a decision system $\mathrm{DS} = (U, C \cup D)$, for each $P \subseteq C$, $K(P) = \{[u_i]_P | u_i \in U$ is called a knowledge of P on U. Each element of $K(P)$ is a class in U/P, also referred as a knowledge granule. Let denote family of all knowledge on U by $\mathcal{K}(U)$. For any $K(P), K(Q) \in \mathcal{K}(U)$, the metric between $K(P), K(Q)$ is defined as

$$d\big(K(P), K(Q)\big) = 1 - \frac{1}{|U|} \sum_{i=1}^{|U|} \frac{|[u_i]_P \cap [u_i]_Q|}{|[u_i]_P \cup [u_i]_Q|}.$$

Proposition 1 [12]. Given a decision system $\mathrm{DS} = (U, C \cup D)$. Note $U/C = \{C_1, C_2, \dots, C_m\}$ and $U/D = \{D_1, D_2, \dots, D_n\}$, then

$$d\big(K(C), K(C \cup D)\big) = 1 - \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{|C_i \cap D_j|^2}{|U||C_i|}.$$

Definition 4 [12]. Given a decision system $\mathrm{DS} = (U, C \cup D)$ and $R \subseteq C$. R is a reduct of C based on metric d if it satisfies two following conditions:

(1) $$d\big(K(R), K(R \cup D)\big) = d(K(C), K(C \cup D));$$

(2) $$\forall r \in R d\big(K(R - \{r\}),\ K(R - \{r\} \cup D)\big) \neq d(K(C),\ K(C \cup D)).$$

Definition 5 [12]. Given a decision system $\mathrm{DS} = (U, C \cup D)$, $B \subseteq C$ and $b \in C - B$. The significance $\mathrm{SIG}_B(b)$ of b with respect to B is

$$\mathrm{SIG}_B(b) = d\big(K(B), K(B \cup D)\big) - d\big(K(B \cup \{b\}), K(B \cup \{b\} \cup D)\big).$$

3. Proposed algorithms for finding the reduct based on metric when the condition attribute set is varied

Proposed results are herein explained in terms of the following notions: Given a decision system $\mathrm{DS} = (U, C \cup D)$. Assume that P is the conditional set which is added or deleted to DS and $U/C = \{C_1, C_2, \dots, C_m\}$; $U/D = \{D_1, D_2, \dots, D_n\}$.

3.1. Finding reduct in decision systems when adding a condition attribute set

When adding P into C, the partition of U by $P \cup C$ is finer than the partition of U by C. We can assume in that case some equivalence classes are unchanged while the others are partitioned into smaller equivalence classes. Note $U/(C \cup P) = \{C_1, C_2, , ... C_k, P_1, P_2, ..., P_l\}$ where P_i is a subset of C_j, $j = k+1, \dots, m$.

Proposition 2. The incremental formula for the metric d after adding condition attribute set P is

$$d\big(K(C \cup P), K(C \cup P \cup D)\big) = d\big(K(C), K(C \cup D)\big) - \Delta,$$

where $\Delta = \frac{1}{|U|}\left(\sum_{i=k+1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{t_i}\frac{|C_i^t \cap D_j|^2}{|C_i^t|} - \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|}\right)$.

Proof: According to Proposition 1,

$$d\big(K(C \cup P), K(C \cup P \cup D)\big) = 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=1}^{l}\sum_{j=1}^{n}\frac{|P_i \cap Y_j|^2}{|P_i|}\right).$$

Assume that each equivalence class C_i, $i = k+1, \dots, m$, is partitioned to t_i classes:

$$C_i = C_i^1 \cup C_i^2 \cup \dots \cup C_i^{t_i}.$$

Then,

$$d\big(K(C \cup P), K(C \cup P \cup D)\big) = 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=1}^{l}\sum_{j=1}^{n}\frac{|P_i \cap Y_j|^2}{|P_i|}\right) =$$

$$= 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|\cup_{t=1}^{t_i} C_i^t \cap D_j|^2}{\left|\cup_{t=1}^{t_i} C_i^t\right|}\right) =$$

$$= 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=k+1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{t_i}\frac{|C_i^t \cap D_j|^2}{|C_i^t|}\right) =$$

$$= 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=k+1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{t_i}\frac{|C_i^t \cap D_j|^2}{|C_i^t|} - \right.$$

$$\left. - \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|}\right) = 1 - \frac{1}{|U|}\left(\sum_{i=1}^{k}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} + \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|}\right) -$$

$$- \frac{1}{|U|}\left(\sum_{i=k+1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{t_i}\frac{|C_i^t \cap D_j|^2}{|C_i^t|} - \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|}\right) = d\big(K(C), K(C \cup D)\big) - \Delta,$$

where

$$\Delta = \frac{1}{|U|}\left(\sum_{i=k+1}^{m}\sum_{j=1}^{n}\sum_{t=1}^{t_i}\frac{|C_i^t \cap D_j|^2}{|C_i^t|} - \sum_{i=k+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|}\right).$$

We show later a simple example to illustrate the incremental formula.

Example 1. Consider DS $= (U, \ C \cup D)$ as in Table 1. P is incremental attribute set consisting of a_1 and a_2,

$$U/D = \{\{x_1, x_5, x_8\}, \{x_2, x_4, x_9\}, \{x_3, x_6, x_7\},$$

$$U/C = \{C_1 = \{x_1, x_4\}, \qquad C_2 = \{x_2, x_3, x_5, x_9\}, C_3 = \{x_6\},$$

$C_4 = \{x_7\}, C_5 = \{x_8\}\}$; $U/(C \cup P) = \{C_1^1 = \{x_1\}, C_1^2 = \{x_4\},$
$C_2^1 = \{x_2, x_9\}, C_2^2 = \{x_3, x_5\},\ C_3, C_4, C_5\}$; $d(K(C), K(C \cup D) = \frac{7}{18}$.

When adding P to C, three classes C_3, C_4, C_5 are unchanged; C_1, C_2 are partitioned to sub classes C_1^1, C_1^2 and C_2^1, C_2^2. Calculate $\Delta = \frac{1}{6}$ thus

$$d\big(K(C \cup P),\quad K(C \cup P \cup D)\big) = \frac{7}{18} - \frac{1}{6} = \frac{2}{9}.$$

Table 1. An example of incremental DS

U	P		C			D
	a_1	a_2	a_3	a_4	a_5	
x_1	1	0	0	1	0	2
x_2	0	1	0	1	1	1
x_3	0	0	0	1	1	0
x_4	0	0	0	1	0	1
x_5	1	0	0	1	1	2
x_6	1	0	0	0	0	0
x_7	1	1	0	0	1	0
x_8	1	0	1	0	0	2
x_9	0	1	0	1	1	1

Base on Definition 8 of significance of attributes and the incremental formula in Proposition 2, we suggest an incremental algorithm for reduct computation as following.

Algorithm 1. A metric-based Incremental Algorithm (IA) for attribute reduction in decision systems when adding a condition attribute set **(IA_MBAR)**

Input: A decision system DS $= (U, C \cup D)$, Core(C) set of C, Reduct set R_C of C and added set P ($P \cap C = \varnothing$)

Output: Reduct R of $C \cup P$

Step 1. Compute

$$U/(C \cup P) =$$
$$= \{C_1,\ C_2,, \dots C_k,\ C_{k+1}^1,\ C_{k+1}^2, \dots,\ C_{k+1}^{t_{k+1}},\ C_{k+2}^1,\ C_{k+2}^2, \dots, C_{k+2}^{t_{k+2}}, \dots,\ C_m^1,\ C_m^2, \dots, C_m^{t_m}\}$$

Step 2. Compute $d\big(K(C \cup P), K(C \cup P \cup D)\big)$

//According to Proposition 2

Step 3. $\mathrm{Core}(P) = \emptyset$

Step 4. For each $a \in P$

Step 5. If $d\big(K(P - \{a\}), K(P - \{a\} \cup D)\big) \neq d(K(P), K(P \cup D))$ then $\mathrm{Core}(P) = \mathrm{Core}(P) \cup \{a\}$

Step 6. $R = R_C \cup \mathrm{Core}(P)$

Step 7. If $d(K(R), K(R \cup D)) = d(K(C \cup P), K(C \cup P \cup \mathrm{D})$ then go to Step 11

Step 8. While $d(K(R), K(R \cup D)) \neq d(K(C \cup P), K(C \cup P \cup D)$ do

Step 9. Begin

For each $a \in C \cup P - R$ compute $\mathrm{SIG}_R(a)$
Select a_m such that $\mathrm{SIG}_R(a_m) = \mathrm{Max}_{a \in C \cup P - R}\{\mathrm{SIG}_R(a)\}$
$R = R \cup \{a_m\}$
End

Step 10. For each $a \in R$ Compute $d(K(R - \{a\}),\ K(R - \{a\} \cup D))$
If $d\big(K(R - \{a\}),\ K(R - \{a\} \cup D)\big) = d(K(C \cup P),\ K(C \cup P \cup D))$
then $R = R - \{a\}$

Step 11. Return R

In Algorithm 1, the time complexity for computing $U/(C \cup P)$ is $O(|C \cup P||U|)$ according to the quick partition algorithm proposed by X u et al. [19]. The time complexity of the incremental metric computation is $O(|U|^2)$; From Steps 3 to 6, time complexity for computing the Core is $O(|P|^2|U|)$. The time complexity for the While loop 8 is $O(|U||C \cup P|^2)$. Step 10 to eliminate dispensable attributes from R has time complexity of $O(|U||C \cup P|)$. So the time complexity of the Algorithm 1 is $O((|U||C \cup P|^2 + |U|^2)$. In case of repetition the Algorithm in [12] for the new condition attribute set $C \cup P$, the time complexity is $O(|U||C \cup P|^2 + |C \cup P||U|^2)$.

In our later experiments, for the data set where $C \cup P$ is not large, time improvement seems little in comparison. However, in the large databases the improvement in time consumption is considerable.

3.2. Finding reduct in decision systems when deleting a condition attribute set

When deleting a set P from C ($P \subset C$), the partition of U by $P - C$ is coarser than the partition of U by C. Assuming that some equivalence classes are unchanged while the others are united from several classes. Note

$$\frac{U}{C-P} = \{C_1,\ C_2, \dots, C_r,\ C^{r+1},\ C^{r+2}, \dots,\ C^k\},$$

where $C^i = \cup\, C_t,\ \ i = r+1, \dots, k;\ \ \ r + 1 \le t \le m;\ k \le m.$

Proposition 3. The incremental formula for the metric d after deleting condition attribute set P from C is:

$$d\big(K(C-P), K(C-P \cup D)\big) = d\big(K(C), K(C \cup D)\big) + \Delta,$$

where $\Delta = \frac{1}{|U|}\left(\sum_{i=r+1}^{m}\sum_{j=1}^{n}\frac{|C \cap Y_j|^2}{|C_i|} - \sum_{i=r+1}^{k}\sum_{j=1}^{n}\frac{|C^i \cap Y_j|^2}{|C^i|}\right).$

P r o o f: Similarly implying as Proposition 2, we have

$$d\big(K(C-P), K(C-P \cup D)\big) = 1 - \frac{1}{|U|}\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D_j|^2}{|C_i|} =$$

$$= 1 - \frac{1}{|U|}\left(\sum_{i=1}^{r}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|} + \sum_{i=r+1}^{k}\sum_{j=1}^{n}\frac{|C^i \cap D_j|^2}{|C^i|}\right) =$$

$$= 1 - \frac{1}{|U|}\left(\sum_{i=1}^{r}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|} + \sum_{i=r+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|} + \sum_{i=r+1}^{k}\sum_{j=1}^{n}\frac{|C^i \cap D_j|^2}{|C^i|} - \right.$$

$$\left. - \sum_{i=r+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|}\right) = 1 - \frac{1}{|U|}\left(\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|}\right) +$$

$$+ \frac{1}{|U|}\left(\sum_{i=r+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|} - \sum_{i=r+1}^{k}\sum_{j=1}^{n}\frac{|C^i \cap D_j|^2}{|C^i|}\right) =$$

$$= d\big(K(C), K(C \cup D)\big) + \Delta,$$

where $\Delta = \frac{1}{|U|}\left(\sum_{i=r+1}^{m}\sum_{j=1}^{n}\frac{|C_i \cap D|^2}{|C_i|} - \sum_{i=r+1}^{k}\sum_{j=1}^{n}\frac{|C^i \cap D_j|^2}{|C^i|}\right)$.

Algorithm 2. A metric-based incremental algorithm for attribute reduction in decision systems when deleting a condition attribute set

Input: A decision system $\mathrm{DS} = (U, C \cup D)$, $\mathrm{Core}(C)$ set of C, reduct set R_C on C and deleted set P

Output: Reduct R of $C - P$

Step 1. $R = R_C$

Step 2. If $P \cap R = \emptyset$ go to Step 9

Step 3. $R = R - P$

Step 4. Compute $U/(C - P) = \{C_1, C_2, \ldots, C_r, C^{r+1}, C^{r+2}, \ldots, C^k\}$. Compute $d\big(K(C - P), K(C - P \cup D)\big)$
//According to Proposition 3

Step 5. If $d(\big(K(R), K(R \cup D)\big) = d(\big(K(C - P), K(C - P \cup D)\big)$ go to Step 9

Step 6. While $d(\big(K(R), K(R \cup D)\big) \neq d(\big(K(C - P), K(C - P \cup D)\big)$ do

Step 7. Begin

For each $a \in C - P - R$ compute $\mathrm{SIG}_R(a)$
Select a_m such that $\mathrm{SIG}_R(a_m) = \mathrm{Max}_{a \in C - P - R}\{\mathrm{SIG}_R(a)\}$
$R = R \cup \{a_m\}$

End

Step 8. For each $a \in R$
Compute $d(K(R - \{a\}),\ K(R - \{a\} \cup D))$
If $d\big(K(R - \{a\}),\ K(R - \{a\} \cup D)\big) = d(K(C - P),\ K(C - P \cup D))$ then $R = R - \{a\}$

Step 9. Return R

By similar inducements, the time complexity for the Algorithm 2 when deleting a condition attribute set P from a decision system is $O(|U|^2 + |C - P|^2|U|)$.

In theory, the two incremental algorithms are not dramatically improved in time consumption. However, in reality, when just some classes are changed during the variant of the condition attribute set, the time improvement is considerable.

4. Experimental analysis

Our experiments aim to two goals: First, to evaluate the performance of Algorithm 1 in comparison to non-incremental algorithm [12]; second, to evaluate the effectiveness of Algorithm 1 in comparison with incremental algorithms for reducts in [17]. The reason we choose proposed algorithms in [17] since using information entropy for attribute reduction is quite well known in research works.

To conduct the experiments, we get six data sets from UCI databases which are briefly described in Table 2. In the data sets, condition attributes will be denoted by 1, 2, 3,… /*C*/. Experiments are carried out on Windows 7 PCs with configuration is Intel(R) Core (TM) i3, CPU (2.66 GHz), 4.00 GB of RAM. Algorithms are programmed with C# on Microsoft Visual Studio 2010.

Table 2. Experimental data files

Order	Data files	Number of objects	Number of condition attributes	Number of classes
1	Hepatitis.data	155	19	3
2	Lung-cancer.data	32	56	4
3	Import-85.data	205	25	6
4	kr-vs-kp.data	3196	36	2
5	Dermatology.data	366	34	6
6	Backup-large.data	307	35	19

4.1. Experiments to evaluate the performance of the proposed incremental algorithms IA_MBAR in comparison with non-incremental Algorithm [12]

For each data set in the Table 2, we carry out followings:

Select 50% condition attributes (from attribute 1 upward) and decision attributes as the start decision system and run the algorithm in [12] on this data for the reduct.

For the last 50% condition attributes, at first time we get 50% attributes as the additional set. Then, we put the whole set (100%) for the second addition. Run the algorithm [12] and IA_MBAR on incremental data for the results showed in the Table 3.

Table 3. Experimental results comparing the MBAR algorithm and IA_MBAR algorithm

Data files	Adding incremental condition attributes, %	Algorithm [12]		IA_MBAR algorithm	
		Reducts	Time, s	Reducts	Time, s
Hepatitis. data	50 100	2, 3, 15 2, 15, 16	0.310 0.327	2, 3, 15 2, 15, 16	0.012 0.015
Lung-cancer. data	50 100	3, 4, 9, 25, 32 3, 4, 9, 43	0.582 0.620	3, 4, 9, 25, 32 3, 4, 9, 43	0.040 0.052
Import-85.data	50 100	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	2.342 2.839	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	0.386 0.422
kr-vs-kp .data	50 100	1- 8, 10- 13, 15-18, 20, 21, 23-27 1, 3- 7, 10- 13, 15- 18, 20-28, 30, 31, 33-36	206.232 219.250	1-8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33- 36	19.254 23.256
Dermato-logy.data	50 100	1-4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	3.286 3.853	1-4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	0.414 0.462
Backup-large.data	50 100	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	1.422 1.840	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.124 0.162

Results in Table 3 show that the reducts are the same for both algorithms but the running time of the incremental algorithm is decreased, especially when the data is large (such as in kr-vs-kp.data). The results illustrate effectiveness and efficiency of the incremental approach for finding reducts.

4.2. Experiments to compare the IA_MBAR algorithm with DIA_RED algorithm using entropies in [17]

The target of the experiments is to evaluate the efficiency of the proposed incremental algorithm IA_MBAR in comparison with the other incremental algorithm DIA_RED in [17] using Liang entropy, Shannon's entropy and combination entropy. In our experiments, six data sets are employed as in Table 2 and similar to the experiments above, at first we choose 50% condition attributes from 1 upward and the decision attributes as the start decision systems. Then we divide the last 50% attributes into two parts as incremental sets and conduct experiments on the new decision system after adding these sets with both DIA_RED and IA_MBAR algorithms.

4.2.1. Results of the experiments to compare the IA_MBAR algorithm with the DIA_RED algorithm using Shannon's entropy

Results in Table 4 show that the two reducts in both algorithms are identical in cases of adding 50 or 100% incremental set. These outcomes clearly illustrate the theoretical results which proved in [12] that Shannon's entropy based reduct is the same as metric based reduct. However, the running time for our proposed incremental algorithm IA_MBAR is less than the running time of the algorithm DIA_RED. This is more obvious in large data (such as in kr-vs-kp.data).

Table 4. Experimental results when running IA_MBAR and DIA_RED using Shannon's entropy

Data files	Adding incremental condition attributes, %	DIA_RED (using Shannon's entropy)		IA_MBAR	
		Reducts	Time, s	Reducts	Time, s
Hepatitis. data	50 100	2, 3, 15 2, 15, 16	0.018 0.022	2, 3, 15 2, 15, 16	0.012 0.015
Lung-cancer. data	50 100	3, 4, 9, 25, 32 3, 4, 9, 43	0.042 0.067	3, 4, 9, 25, 32 3, 4, 9, 43	0.040 0.052
Import-85.data	50 100	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	0.426 0.612	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	0.386 0.422
kr-vs-kp .data	50 100	1-8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	22.254 28.256	1-8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	19.254 23.256
Dermato-logy.data	50 100	1-4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	0.418 0.486	1-4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	0.408 0.462
Backup-large.data	50 100	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.186 0.210	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.124 0.162

Table 5. Experimental results when running IA_MBAR and DIA_RED using Liang's entropy in [17]

Data files	Adding incremental condition attributes, %	DIA_RED using Liang's entropy (complementary entropy)		IA_MBAR	
		Reducts	Time, s	Reducts	Time, s
Hepatitis. data	50 100	2, 3, 15 2, 15, 16	0.010 0.016	2, 3, 15 2, 15, 16	0.012 0.015
Lung-cancer. data	50 100	3, 4, 9, 25, 32, 38, 41 3, 4, 9, 43, 46, 54	0.038 0.050	3, 4, 9, 25, 32 3, 4, 9, 43	0.040 0.052
Import-85.data	50 100	1, 2, 6, 7, 11, 14, 15, 16 1, 2, 7, 14, 18, 20, 21	0.410 0.432	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	0.386 0.422
kr-vs-kp .data	50 100	1- 8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	20.186 24.128	1-8, 10-13, 15-18, 20, 21, 23- 27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	19.254 23.256
Dermato-logy.data	50 100	1-5, 9, 14, 15, 19, 22 1, 4, 5, 9, 15, 22, 33, 34	0.398 0.406	1-4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	0.408 0.462
Backup-large.data	50 100	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.126 0.174	1-4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.124 0.162

4.2.2. Results of the experiments to compare the IA_MBAR algorithm with the DIA_RED algorithm using Liang's entropy (complementary entropy)

Results in Table 5 show that, for the data set Hepatitis.data, kr-vs-kp.data two algorithms give the same reducts. For the last data sets, the reducts obtained by our algorithm have smaller cardinality than those ones obtained by DIA_RED. This experimental results illustrate for the theoretically proof in [18]. For the running time, it can be considered as the same for both algorithms.

4.2.3. Results of the experiments to compare the IA_MBAR algorithm with the DIA_RED algorithm using combination entropy

Table 6. Experimental results when running IA_MBAR and DIA_RED using combination entropy in [17]

Data files	Adding incremental condition attributes, %	DIA_RED (using combination entropy)		IA_MBAR	
		Reducts	Time, s	Reducts	Time, s
Hepatitis.data	50 100	2, 3, 15 2, 15, 16	0.008 0.014	2, 3, 15 2, 15, 16	0.012 0.015
Lung-cancer. data	50 100	3, 4, 9, 25, 32, 38, 41 3, 4, 9, 43, 46, 54	0.034 0.054	3, 4, 9, 25, 32 3, 4, 9, 43	0.040 0.052
Import-85.data	50 100	1, 2, 6, 7, 11, 14, 15, 16 1, 2, 7, 14, 18, 20, 21	0.398 0.424	1, 2, 7, 11, 14, 16 1, 2, 7, 14, 20, 21	0.386 0.422
kr-vs-kp.data	50 100	1-8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	18.258 22.568	1-8, 10-13, 15-18, 20, 21, 23-27 1, 3-7, 10-13, 15-18, 20-28, 30, 31, 33-36	19.254 23.256
Dermatology. data	50 100	1-5, 9, 14, 15, 19, 22 1, 4, 5, 9, 15, 22, 33, 34	0.386 0.428	1- 4, 9, 14, 15, 22 1, 4, 9, 15, 22, 33, 34	0.408 0.462
Backup-large.data	50 100	1- 4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.118 0.168	1- 4, 6, 7, 8, 10, 16, 22 1, 3, 4, 6, 7, 8, 10, 16, 29	0.124 0.162

The results in Table 6 show that, the data set Hepatitis.data, kr-vs-kp.data and Backup-large.data have the same reducts. For the last four data files, the reducts obtained by our algorithm have smaller cardinality than those ones obtained by DIA_RED.

4.3. Result discussions

For the experimental results gaining above, we have some discussions as followings:

Our incremental algorithm IA_MBAR is much more effective in running time in comparison with the non-incremental algorithm in [12].

To compare with others incremental algorithms, we choose entropy method group [17] since these algorithms are quite popular recently as mentioned in the introduction of this paper. From the experimental results showed in Tables 4, 5, 6 we can conclude that in the case of incremental Shannon's entropy based reduction our reducts is better in terms of running time. For the last two situations of using Liang's and combination entropy, our method give the better reducts in terms of the number of attributes. Moreover, these results also consolidate the theoretical research in [18] so that to choose our method for gaining reducts in dynamic decision systems when condition attribute set varied could be considered, especially when the data is large.

5. Conclusions

In the real world today, databases usually grow, change and update. To propose effective methods in order to optimize time consumption for carrying algorithms finding reducts as well as get the best reduct is a goal of researchers. In this paper, we construct mechanisms for updating the metric used in attribute reduction algorithms and propose two heuristic algorithms to work out the reducts in the case of adding or deleting a condition attribute set from decision systems. Since suggested algorithms are not re-implemented on the varied database so that we improve calculating time for finding reducts. Experiments on UCI data sets show that the algorithms are effectively reduce consuming time in comparison with apply directly the non-incremental algorithm for the varied decision systems. We also do comparisons to other incremental method proposing in [17], the results show that our method are better in terms of the number of attributes in reducts. Furthermore, we have dealt with the case of deleting a set of condition attributes which is not solved in the [17]. Besides, employing distance in difference measuring is well known for its simplicity and effectiveness and we gained that in this work. Our next step in research could be using metric to constructing algorithms for finding reducts in case of multi objects varied.

References

1. S l e z a k, D. Degrees of Conditional (in) Dependence: A Framework for Approximate Bayesian Networks and Examples Related to the Rough Setbased Feature Selection. – Information Sciences, Vol. **179**, 2009, pp. 197-209.
2. D e m e t r o v i c s, J., V. D. T h i, N. L. G i a n g. Metric Based Attribute Reduction in Dynamic Decision Tables. – Annales Univ. Sci. Budapest, Sect. Comp. Vol. **42**, 2014, pp. 157-172.

3. D e m e t r o v i c s, J., V. D. T h i, N. L. G i a n g. An Efficient Algorithm for Determining the Set of All Reductive Attributes in Incomplete Decision Tables. – Cybernetics and Information Technologies, Vol. **13**, 2013, No 4, pp. 118-126.
4. D e m e t r o v i c s, J., V. D. T h i, N. L. G i a n g . On Finding All Reducts of Consistent Decision Tables. – Cybernetics and Information Technologies, Vol. **14**, 2014, No 4, pp. 3-10.
5. W a n g, F., J. L i a n g, C. D a n g. Attribute Reduction for Dynamic Data Sets. – Applied Soft Computing, Vol. **13**, 2013, No 1, pp. 676-689.
6. W a n g, G. Y., H. Y u, D. C. Y a n g. Decision System Reduction Based on Conditional Information Entropy. – Chinese Journal of Computer, Vol. **25**, 2002, No 7, pp. 759-766.
7. G i a n, L. H. An Incremental Updating Algorithm of Attribute Reduction Set in Decision Tables. – In: Proc. of 6th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD'09, Vol. **2**, 2009, pp. 421-425.
8. H u, F., G.Y. W a n g, H. H u a n g, Y. W u. Incremental Attribute Reduction Based on Elementary Sets. – In: Proc. of 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Regina, Canada, 2005, pp. 185-193.
9. L i a n g, J., R. L i, Y. Q i a n. Distance: A More Comprehensible Perspective for Measures in Rough Set Theory. – Knowledge-Based Systems, Vol. **27**, 2012, pp. 126-136.
10. L i a n g, J. Y., K. S. C h i n, C. Y. D a n g, R. C. M. Y a m. New Method for Measuring Uncertainty and Fuzziness in Rough Set Theory. – International Journal of General Systems, Vol. **31**, 2002, pp. 331-342.
11. L i a n g, J. Y., F. W a n g , C. Y. D a n g, Y. H. Q i a n. A Group Incremental Approach to Feature Selection Applying Rough Set Technique. – IEEE Transactions on Knowledge and Data Engineering, Vol. **26,** 2014, No 2, pp. 294-308.
12. L o n g, G. N. Metric Based Attribute Reduction in Decision Tables. – In: Proc. of 2012 International Workshop on Rough Sets Applications (RSA'2012), FedCSIS, IEEE, 2012, pp. 333-338.
13. P a w l a k, Z. Rough Sets: Theoretical Aspects of Reasoning About Data. Kluwer Academic Publishers, 1991.
14. V u, D. T., N. L. G i a n g. A Method to Construct Decision Table from Relation Scheme. – Cybernetics and Information Technologies, Vol. **11**, 2011, No 3, pp. 32-41.
15. V u, D. T., N. L. G i a n g. A Method for Extracting Knowledge from Decision Tables in Terms of Functional Dependencies. – Cybernetics and Information Technologies, Vol. **13**, 2013, No 1, pp. 73-82.
16. Q i a n, W., W. S h u, B. Y a n g, C. Z h a n g. An Incremental Algorithm to Feature Selection in Decision Systems with the Variation of Feature Set. – Chinese Journal of Electronics, Vol. **24**, 2015, No 1.
17. W a n g, F., J. Y. L i a n g, Y. H. Q i a n. Attribute Reduction: A Dimension Incremental Strategy. – Knowledge-Based Systems**,** Vol. **39**, 2013, pp. 95-108.
18. Q i a n, Y. H., J. Y. L i a n g, D. Y. L i, H. Y. Z h a n g, C. Y. D a n g. Measures for Evaluating the Decision Performance of a Decision Table in Rough Set Theory. – Information Sciences, Vol. **178**, 2008, pp.181-202.
19. X u, Z. Y., Z. P. L i u, B. R. Y a n g, W. S o n g. A Quick Attribute Reduction Algorithm with Complexity of Max ($O(|C||U|)$, $O(|C|^2|U/C|)$). – Journal of Computer, Vol. **29**, 2006, No 3, pp. 391-398.
20. Z h a n g, C. S., J. J. R u a n, Y. H. T a n. An Improved Incremental Updating Algorithm for Core Based on Positive Region. – Journal of Computational Information Systems, Vol. **7**, 2011, No 9, pp. 3127-3133.