Feature Selection using Compact Discernibility Matrix-based Approach in Dynamic Incomplete Decision System

WENBIN QIAN[1,2], WENHAO SHU[3], YONGHONG XIE[1,2,*], BINGRU YANG[1,2]

[1]*School of Computer and Communication Engineering, University of Science and Technology Beijing, 100083, China*

[2]*Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing, 100083, China*

[3]*School of Computer and Information Technology, Beijing Jiaotong University, Beijing, 100044, China*

E-mail: {qianwenbin1027, xyzwlx, bryang_kdd}@126.com, 11112084@bjtu.edu.cn

According to whether the systems vary over time, the decision systems can be divided into two categories: static decision systems and dynamic decision systems. Most existing feature selection work is done for the former, few work has been developed recently for the latter. To the best of our knowledge, when an object set varies dynamically in incomplete decision systems, no feature selection approach has been specially designed to select feature subset until now. In this regard, a feature selection algorithm based on compact discernibility matrix is developed. The compact discernibility matrix is firstly introduced, which not only avoids computing the time-consuming lower approximation, but also saves more storage space than classical discernibility matrix. Afterwards, we take the change of lower approximation as a springboard to incrementally update the compact discernibility matrix. On the basis of updated compact discernibility matrix, an efficient feature selection algorithm is provided to compute a new feature subset, instead of retaining the discernibility matrix from scratch to find a new feature subset. The efficiency and effectiveness of the proposed algorithm are demonstrated by the experimental results on different data sets.

Keywords: Feature selection, Lower approximation, Dynamic incomplete decision system, Compact discernibility matrix, Rough sets

1. INTRODUCTION

Feature selection is necessary involved data sets containing huge numbers of features in pattern recognition, data mining and machine learning [9, 27, 31]. Excessive irrelevant features to knowledge discovery tasks may lead to increase the processing time and deteriorate the classification performance [3, 4, 20]. The selection of relevant and significance features is an essential preprocessing step particularly for data sets with thousands of features. As we known, feature selection is an important task to acquire compact decision rules before decision-making analysis. Many feature selection methods have been proposed. They can be roughly divided into two categories: wrapper [27] and filter [36] approaches. The wrapper methods

Recevied: July, 28, 2013 revised: January, 12, 2014
Communicated by

*This work was partially supported by the National Natural Science Foundation of China (No.61175048); the Key Project of Ministry of Science and Technology of China (No.2010IM020900); the 2012 Ladder Plan Project of Beijing Key Laboratory of Knowledge Engineering for Materials Science (No.Z121101002812005).

employ a learning algorithm to evaluate the goodness of selected feature subsets based on the predictive accuracy. However, the filter methods are independent of learning algorithms to evaluate the significance of features. Rough set-based feature selection does not need any preliminary or additional information about data such as probability distributions in statistics, basic probability assignments in Dempster-Shafer theory [34], or a grade of membership in fuzzy set theory [28, 33]. Rough sets proposed by Pawlak [5, 6] provides a theoretical foundation for data analysis, which has been widely applied in many areas such as artificial intelligence and knowledge acquisition [7-11]. The objective of feature selection is to select a minimal feature subset that can provide the same information for classification as the full original set of available features.

Regarding feature selection, a great number of techniques have been designed in the framework of rough set theory until now. The techniques can be roughly divided into three groups: positive region [30], information entropy [24, 25, 32] and discernibility matrix [2, 6, 22, 23]. Pawlak's classical feature selection is intended to apply in complete decision systems. However, due to various factors such as noise in the data, lack of critical information and prediction capability, some missing values are possibly occurred in an object on some condition features. Such decision systems are regarded as incomplete decision systems (IDSs), in which any missing feature value is represented by special symbol "*". In fact, we are often faced with the IDSs in many practical applications [1, 12, 14, 19]. There is no guarantee for classical feature selection to preserve useful in the IDSs. The key reason is that the equivalence relation in complete decision systems is not applicable to the IDSs. From the viewpoint of information theory, Qian et al. developed a combination entropy-based feature selection algorithm to select a feature subset [26]. Luo and Yang presented a limited dominance-based knowledge reduction approach [18]. Meng et al. proposed a positive region-based feature selection algorithm [15]. Kryszkiewicz proposed one of the pioneering work employed discernibility matrix in incomplete decision systems, all the feature subsets can be derived from the constructed discernibility function [1]. Later, Leung and Li provided a maximal consistent block technique to create a simpler discernibility function, and acquired the rules from consistent incomplete decision systems [12]. Qian et al. applied the maximal consistent block technique to construct two discernibility matrices for lower and upper approximation feature selection in incomplete decision systems [19]. Among these feature selection algorithms based on discernibility matrix, they are intuitive and concise. However, numbers of empty matrix elements appear in the discernibility matrix may lead to waste in both the storage space and the computational time. To overcome the deficiency, a compact discernibility matrix is constructed in this paper. On the basis of the above analyses, all the feature selection algorithms focus on the static incomplete decision systems.

As to dynamic characteristics of data collection, the feature set, object set and feature values in decision systems all evolve over time. With respect to dynamic decision systems, incremental learning approach has been demonstrated to be effective and efficient, which can avoid some redundant computations. When an object set adds into or deletes from complete decision systems, Xu et al. designed an incremental feature selection algorithm based on integer programming to find the new feature subset [13]. When a single object adds into or deletes from a complete decision system, Yang et al. developed an incremental feature selection algorithm based on improved discernibility matrix [17]. However, there are many empty elements in the matrix, which occupy much storage space and computational time. Shu and Shen developed an efficient updated feature selection algorithm for incomplete decision systems with dynamically varying feature set [40]. When a single object adds into or deletes from complete decision systems, Hu et al. designed an incremental feature selection algorithm to find a new feature subset [21]. However, the algorithm can not simply transplant to incomplete decision systems. In addition, when an objects set adds into or deletes from complete decision systems, Liu et al. developed an incremental model and approach as well as its algorithm for inducing interesting knowledge [16]. Wang et al. developed an incremental feature selection algorithm for complete decision systems with dynamically varying data values [41]. Different from above algorithms, we discuss the feature selection work employed a tolerance relation in incomplete decision systems with the variation of an object set.

Among various heuristic feature selection algorithms, the lower approximation-based feature selection algorithm has been extensively researched in rough set theory. One can extract knowledge hidden in data that represented as certain rules. The key work of lower approximation-based feature selection algorithm is to compute the tolerance classes in incomplete decision systems. A classical algorithm for calculating the tolerance classes with the time complexity of $O(|C||U|^2)$ [1], where $|C|$ and $|U|$ denotes the number of features and objects, respectively. Using this computation of tolerance classes, the lower approximation-based feature selection algorithm [26, 33] costs no less than $O(|C|^3|U|^2)$ to find a feature subset. To avoid this not computationally costless work, we consider an equivalent method based on discernibility matrix to compute a feature subset in incomplete decision systems. Different from the classical discernibility matrices with the storage space of $O(|C||U|^2)$[2, 6, 22, 23], here we propose a compact discernibility matrix, in which only non-empty matrix elements that appear in the matrix are captured. In this way, both storage space and computational time are saved to complete the feature selection tasks. On the basis of compact discernibility matrix, we just update the matrix incrementally to compute a new feature subset in a dynamic incomplete decision system, which is more efficient than retraining the compact discernibility matrix from scratch.

The remainder of this paper is organized as follows. In Section 2, we review some basic concepts and relevant definitions involved in this paper. In Section 3, some theoretical foundations are provided to associate an equivalent feature selection approach based on compact discernibility matrix. In Section 4, we firstly take the change of lower approximation as a springboard to incrementally update the compact discernibility matrix, then an efficient feature selection algorithm is developed. Experiments are presented to validate the efficiency and effectiveness of the proposed algorithm in Section 5. Section 6 concludes the paper.

2. PRELIMINARIES

In this section, some basic concepts and relevant definitions in this paper are briefly reviewed, which make preparations for further discussions.

2.1. Basic concepts

An information system is expressed as a quadruple $IS = (U, A, V, f)$, where,

(1) U is a non-empty finite set of objects, called the universe of course;

(2) A is a non-empty finite set of features;

(3) V is the union of feature domains, $V = \bigcup_{a \in A} V_a$, where V_a is the value set of feature a, indicating the domain of a;

(4) $f : U \times A \to V$ is an information function which assigns particular values from objects to domains of feature such as $\forall a \in A,\ x \in U,\ f(x,a) \in V_a$, where $f(x,a)$ denotes the value of feature a for object x.

For any feature subset $P \subseteq A$, P determines a binary relation on U, denoted by $IND(P)$, which is defined as follows: $IND(P) = \{(x,y) \in U \times U | \forall a \in P, f(x,a) = f(y,a)\}$. It can be easily known that $IND(P)$ is an equivalence relation and it constructs a partition of U, denoted by $U/IND(P)$. The equivalence class containing x is denoted by $[x]_P = \{y | (x,y) \in IND(P)\}$.

If there exist $x \in U$ and $a \in A$ such that $f(x,a)$ equals to a missing value (a null of unknown value, denoted as “ * ”), which means for at least one feature $a \in A$, $* \in V_a$. Then the information system is called an incomplete information system (IIS). Otherwise, it is called a complete information system (CIS).

A decision system (DS) is expressed as $DS = (U, C \cup D, V, f)$, where,

(1) C and D denote the set of condition features and the set of decision features, respectively, $C \cap D = \varnothing$;

(2) V is the union of feature domain, $V = V_C \cup V_D = \{V_a | a \in C\} \bigcup \{V_d | d \in D\}$.

If $* \notin V_D$, but $* \in V_C$, then the decision system is called an incomplete decision system (IDS). If $* \notin V_C$ and $* \notin V_D$, then it is called a complete decision system(CDS).

Given an incomplete information system $IIS = (U, A, V, f)$, for any subset of features $P \subseteq A$, P determines a binary relation $TR(P)$ on U as follows: $TR(P) = \{(x,y)|f(x,a) = f(y,a)$ or $f(x,a) = *$ or $f(y,a) = *$, for $\forall a \in P$ and $x,y \in U\}$. It is easily known that $TR(P)$ is reflexive, symmetric and intransitive. Obviously, it is a tolerance relation. For any object $x \in U$, $TR(P)$ determines the maximal set of objects that are possibly indistinguishable to x with respect to P, which is denoted by $S_P(x)$. Equivalently, $S_P(x) = \{y|(x,y) \in TR(P), y \in U\}$. And $U/TR(P)$ denote the family set $\{S_P(x)|x \in U\}$, which is the classification induced by P. Any element from $U/TR(P)$ will be called a tolerance class. Clearly, the tolerance classes of all the objects from U do not constitute a partition. They form a cover of U, i.e., $S_P(x) \neq \emptyset$ for every $x \in U$, and $\bigcup_{x \in U} S_P(x) = U$. And for $X \in U/TR(P)$ is consistent iff all the objects in X have the same decision value; otherwise, X is inconsistent.

Given any subset of features $P \subseteq A$ and subset $X \subseteq U$, the lower and upper approximations of X in terms of tolerance relation $TR(P)$ are defined as

$$\underline{apr_P}X = \{x \in U | S_P(x) \subseteq X\}$$

and

$$\overline{apr}_P X = \{x \in U | S_P(x) \cap X \neq \phi\}.$$

The lower approximation is called the positive region, that is $POS_P(X) = \underline{apr_P}X$. The objects in $\underline{apr_P}X$ belong to X certainly.

Let $IDS = (U, C \cup D, V, f)$ be an incomplete decision system and $P \subseteq C$, the objects are partitioned into r mutually exclusion crisp subsets $U/IND(D) = \{D_1, D_2, \cdots, D_r\}$ by the decision features D. The lower and upper approximations with respect to P of the decision features D are defined as

$$\underline{apr_P}D = \{\underline{apr_P}(D_1), \underline{apr_P}(D_2), \cdots, \underline{apr_P}(D_r)\}$$

and

$$\overline{apr}_P D = \{\overline{apr}_P(D_1), \overline{apr}_P(D_2), \cdots, \overline{apr}_P(D_r)\}.$$

Denoted by $POS_P(D) = \bigcup_{i=1}^{r} \underline{apr_P}(D_i)$, which is called the positive region of D with respect to the condition feature set P in the IDS.

2.2. Feature selection in incomplete decision systems

Using tolerance relation, two definitions of feature selection in incomplete decision systems are shown as follows.

Definition 1. Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, for $P \subseteq C$. P is a selected feature subset in the incomplete decision system iff $POS_P(D) = POS_C(D)$ and $POS_{P'}(D) \neq POS_C(D)$ for any $P^{'} \subset P$.

This definition keeps the lower approximation of target decision unchanged. The first one ensures that lower approximation of target decisions is preserved by a selected feature subset. Therefore, the feature subset is sufficient for preserving the positive region of target decisions. The second condition ensures that the selected feature subset is the minimum, i.e., each feature in the feature subset is necessary.

To bypass the time-consuming computations of positive region, we consider constructing a discernibility matrix to complete feature selection tasks. Because a large number of empty elements that appear in the matrix could result in massive cost to both storage space and computational time. Therefore, we construct a compact discernibility matrix of the incomplete decision system as follows.

Definition 2. Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, and $U/IND(D) = \{D_1, D_2, \cdots, D_r\}$, the compact discernibility matrix M consists of $r-1$ sub-discernibility matrices, a matrix element $m(i,j)$ of each sub-discernibility matrix is defined as follows.

$$m(i,j) = \begin{cases} \{a \in C : f(x_i,a) \neq f(x_j,a) \wedge f(x_i,a) \neq * \wedge f(x_j,a) \neq *\}, \\ \quad when\, x_i \in D_v \wedge x_j \in D_w, 1 \leq v < w \leq r, \\ \phi \qquad else. \end{cases}$$

The matrix element $m(i,j)$ means that the objects x_i and x_j can be discerned by any feature in $m(i,j)$. Obviously, only the objects x_i and x_j that belong to different decision classes need to be discerned. For each sub-discernibility matrix, only the non-empty matrix elements are captured, rather than all elements with space complexity of $O(|C||U|^2)$ in the classical discernibility matrix [2, 6, 22], which save much storage space of matrix. Based on the compact discernibility matrix, the definition of feature selection is defined as follows.

Definition 3. Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, a compact discernibility matrix M is constructed by Definition 2, for $\forall P \subseteq C$. P is a selected feature subset in the incomplete decision system iff $\forall (x_i, x_j) \in U \times U$, $\phi \neq m(i,j) \in M$, such that $P \cap m(i,j) \neq \phi$; and for $\forall a \in P$, $\exists (x_i, x_j) \in U \times U$, $P' = P - \{a\}$, such that $P' \cap m(i,j) = \phi$.

The first condition indicates that the feature subset P is sufficient to preserve the classification ability which is identical to that of the whole feature set C. The second condition means that each feature in P is individually indispensable for preserving the classification power.

3. FEATURE SELECTION BASED ON COMPACT DISCERNIBILITY MATRIX

In this section, to avoid computing the time-consuming lower approximation-based feature selection in incomplete decision systems, we firstly provide some theoretical foundations to associate an equivalent feature selection based on the compact discernibility matrix. To further reduce the computational time, two elementary matrix operations are then given on the compact discernibility matrix before selecting a feature subset.

Lemma 1. *Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, a compact discernibility matrix M is constructed by Definition 2, and $U/IND(D) = \{D_1, D_2, \cdots, D_r\}$, for $\forall P \subseteq C$, if $\forall \phi \neq m(i,j)$, such that $P \cap m(i,j) \neq \phi$, then $POS_P(D) = POS_C(D)$.*

Proof. Suppose $POS_P(D) \neq POS_C(D)$, there at least exists $x_i \in POS_C(D)$ such that $S_C(x_i) \subseteq D_i (1 \leq i \leq r)$ and $S_P(x_i) \nsubseteq D_i$. Because $S_P(x_i) \nsubseteq D_i$, it is obvious that there is $x_j \in S_P(x_i)$ such that $f(x_i, D) \neq f(x_j, D)$. And because $S_C(x_i) \subseteq D_i$ and $x_i \in POS_C(D)$, there exists feature $c_k \in C - P$ such that $f(x_i, c_k) \neq f(x_j, c_k) \wedge f(x_i, c_k) \neq * \wedge f(x_j, c_k) \neq *$, according to the definition of compact discernibility matrix, it holds that $c_k \in m(i,j) \Rightarrow m(i,j) \neq \phi$. Thus there is $P \cap m(i,j) = \phi$, which contradicts with the above condition $P \cap m(i,j) \neq \phi$, so the hypothesis does not hold. Therefore, we have $POS_P(D) = POS_C(D)$. □

Lemma 2. *Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, the corresponding discernibility matrix M is constructed by Definition 2, and $U/IND(D) = \{D_1, D_2, \cdots, D_r\}$, for $\forall P \subseteq C$, if $POS_P(D) = POS_C(D)$, then $\forall \phi \neq m(i,j)$, $P \cap m(i,j) \neq \phi$ holds.*

Proof. For $\forall P \subseteq C$, suppose there exists $\forall \phi \neq m(i,j)$, $P \cap m(i,j) = \phi$ holds, then there at least exists one feature $c_k \in C \wedge c_k \notin P$ such that $c_k \in m(i,j)$. According to the definition of compact discernibility matrix, it is obvious that $x_j \in S_P(x_i)$. And because $x_i, x_j \in POS_C(D)$ and $c_k \in m(i,j)$, it holds that $f(x_i, D) \neq f(x_j, D)$. Thus there are $x_j \in S_P(x_i) \wedge f(x_i, D) \neq f(x_j, D)$, it is obvious that $S_P(x_i) \nsubseteq D_i (1 \leq i \leq r)$, i.e., $x_i \notin POS_P(D)$. Due to $x_i \in POS_C(D)$, thus $POS_P(D) \neq POS_C(D)$ holds, which contradicts with the above condition $POS_P(D) = POS_C(D)$, so the hypothesis does not hold. Therefore, we have $\forall \phi \neq m(i,j)$, $P \cap m(i,j) \neq \phi$. □

From the above two lemmas, we easily get the following theorem to prove the relationships between the compact discernibility matrix-based feature selection and the lower approximation-based feature selection.

Theorem 1. *The feature subset obtained by Definition 3 is equivalent to the feature subset obtained by Definition 1.*

Proof. It follows directly from Lemmas 1 and 2. □

Theorem 1 shows that the same results of feature subsets holds for compact discernibility matrix-based feature selection and lower approximation-based feature selection. The advantages of this equivalent relationship are as follows: one is to avoid computing the time-consuming lower approximation, the other is to take the change of lower approximation as a springboard to incrementally update the compact discernibility matrix in dynamic incomplete decision systems. To further reduce the computational time, two elementary matrix operations are given on the compact discernibility matrix before selecting a feature subset as follows.

Proposition 1. *Given an incomplete decision system IDS* $= (U, C \cup D, V, f)$*, M is the compact discernibility matrix of IDS, for* $\forall m(i,j) \neq \phi$*, if the matrix element is a singleton feature, then it should be reserved in the matrix M.*

Proof. For $\forall m(i,j) \neq \phi$, by the definition of compact discernibility matrix, it means that there are at least one condition features can be used to discern the objects x_i and x_j. If the matrix element is a singleton feature, it means that there is only one condition feature used to discern the pair of objects (x_i, x_j), i.e., the condition feature is indispensable. Therefore, it should be reserved in the matrix M. □

Proposition 2. *Given an incomplete decision system IDS* $= (U, C \cup D, V, f)$*, M is the compact discernibility matrix of IDS, for* $\forall m(i,j) \neq \phi$*, if there exists another matrix element P, such that* $\phi \neq P \subset m(i,j)$*, then the matrix element* $m(i,j)$ *is absorbed by P.*

Proof. For $\phi \neq P \subset m(i,j)$, by this assumption, the pair of objects (x_i, x_j) can be discerned by the matrix element $P = P \cap m(i,j)$. There are two cases for matrix element P. If the matrix element P is a singleton feature, from Proposition 1, it follows that the element P should be reserved in the matrix M. This means that the matrix element $m(i,j)$ is absorbed by P. If the matrix element P is not a singleton feature, suppose the pair of objects (x_i, x_j) can be discerned by one feature from P, since $P \subset m(i,j)$, consequently, the same feature from $m(i,j)$ can also be used to discern the pair of objects (x_i, x_j), which means other features from $m(i,j) - P$ do not need to be considered. Thus it also follows that the matrix element $m(i,j)$ is absorbed by P. This completes the proof. □

To improve the computational efficiency of compact discernibility matrix-based feature selection algorithm, the two above elementary matrix operations may be considered simultaneously. Proposition 1 indicates a singleton feature must contain in the selected feature subset. Proposition 2 denotes the matrix elements that are supersets of a feature subset can be deleted.

4. AN EFFICIENT INCREMENTAL FEATURE SELECTION ALGORITHM BASED ON COMPACT DISCERNIBILITY MATRIX

When an object set immigrates into the incomplete decision system, if reconstructing compact discernibility matrix to compute a new feature subset, it is often time consuming. To overcome this shortcoming, an efficient updated feature selection algorithm is provided to compute new feature subset by only renewing the discernibility matrix in an incremental manner. We first take the change of lower approximation as a springboard to judge whether renewing the original compact discernibility matrix as follows.

4.1. Updating scheme for renewing the compact discernibility matrix

Given an incomplete decision system $IDS = (U, C \cup D, V, f)$, where $U = \{x_1, x_2, \ldots, x_m\}$, $U/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S_C(x_m)\}$ and $U/\,IND(D) = \{D_1, D_2, \ldots, D_r\}$, suppose the original feature subset is P, original lower approximation is $POS_C^U(D)$ and original lower approximation under P is $POS_P^U(D)$. If a new object immigrates into the system, the universe U becomes $U' = U \cup \{x\}$. There are four cases with regard to a new immigration as follows.

Case 1. Forming a new tolerance class and a new decision class.

In this case, $U'/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S_C(x_m), S_C(x)\}$, where $S_C(x) = \{x\}$. In addition, $U'/IND(D) = \{D_1, D_2, \ldots, D_r, D_{r+1}\}$, where $D_{r+1} = \{x\}$. Obviously, it follows that $S_C(x) \subseteq D_{r+1}$, thus $POS_C^{U'}(D) = POS_C^U(D) \cup \{x\}$.

Proposition 3. *(1) If $|S_C(x)| = 1 \wedge |D_{r+1}| = 1 \wedge |S_P(x)| = 1$, then P is also the new feature subset of the changed incomplete decision system.*

(2) If $|S_C(x)| = 1 \wedge |D_{r+1}| = 1 \wedge |S_P(x)| > 1$, then P is not the new feature subset of changed incomplete decision system.

Proof. (1) Since $|S_P(x)| = 1$ and $S_C(x) \subseteq D_{r+1}$, it follows that $S_P(x) \subseteq D_{r+1}$. Thus there is $POS_P^{U'}(D) = POS_P^U(D) \cup \{x\}$. Because $POS_C^{U'}(D) = POS_C^U(D) \cup \{x\}$, and P is the original feature subset, $POS_P^U(D) = POS_C^U(D)$. Therefore, $POS_P^{U'}(D) = POS_C^{U'}(D)$ holds. By Definition 1, it is necessary to further prove that $POS_{P'}^{U'}(D)(D) \neq POS_P^{U'}(D)$ for any $P' \subset P$. Because P is the original feature subset, $POS_{P'}^U(D) \neq POS_P^U(D)$ for any $P' \subset P$. Since $P' \subset P$, it follows that $POS_{P'}^U(D) \subset POS_P^U(D)$, and there is $POS_{P'}^{U'}(D) \subseteq POS_{P'}^{U}(D) \cup \{x\}$. Therefore, $POS_{P'}^{U'}(D) \subset POS_{P}^{U}(D) \cup \{x\} = POS_{P}^{U'}(D)$, so $POS_{P'}^{U'}(D) \neq POS_{P}^{U'}(D)$ holds for any $P' \subset P$. Therefore, P is also the new feature subset of the changed incomplete decision system. (2) Since $|S_P(x)| > 1$, there are more than one objects in $S_P(x)$ form tolerance relation with the new object x under P. Because the tolerance relation is symmetric, it holds that $S'_P(x_i) = S_P(x_i) \cup \{x\}$. Since $D_{r+1} = \{x\}$, both $S_P(x) \nsubseteq D_{r+1}$ and $S_P(x_i) \nsubseteq D_k(1 \leq k \leq r+1)$ hold, where $S'_P(x_i)$ is a changed tolerance class. Thus it follows that $POS_{P}^{U'}(D) = POS_{P}^{U}(D) - S_P(x)$, but we have that $POS_{C}^{U'}(D) = POS_{C}^{U}(D) \cup \{x\}$ and $POS_{P}^{U}(D) = POS_{C}^{U}(D)$, this proves that $POS_{P}^{U'}(D) \neq POS_{C}^{U'}(D)$. Therefore, P is not the new feature subset of the changed incomplete decision system. □

For situation (1), we do not need to update the original compact discernibility matrix, just keep the original feature subset unchanged. However, for situation (2), we need to update the original discernibility matrix in an incremental manner to compute a new feature subset. The new compact discernibility matrix composed of r sub-discernibility matrices is constructed as follows: for $r-1$ original sub-discernibility matrices, we just add the new object x to the last column of each original sub-discernibility matrix, and modify the corresponding row; for other matrix elements, we keep them unchanged; and a new sub-discernibility matrix is generated between the new object x and the objects in the last decision class.

Case 2. Only forming a new tolerance class

In this case, $U'/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S_C(x_m), S_C(x)\}$, where $S_C(x) = \{x\}$. In addition, $U'/IND(D) = \{D_1, D_2, \ldots, D'_j, \ldots, D_r\}$, where $D'_j = D_j \cup \{x\}(1 \leq j \leq r)$. Obviously, it follows that $S_C(x) \subseteq D'_j$, thus $POS_{C}^{U'}(D) = POS_{C}^{U}(D) \cup \{x\}$.

Proposition 4. *(1) If $|S_C(x)| = 1 \wedge S_P(x) \subseteq D_k(1 \leq k \leq r)$, then P is also the new feature subset of the changed incomplete decision system.*

(2) If $|S_C(x)| = 1 \wedge S_P(x) \nsubseteq D_k(1 \leq k \leq r)$, then P is not the new feature subset of the changed incomplete decision system.

Proof. (1) Since $S_P(x) \subseteq D_k$, it follows that $POS_{P}^{U'}(D) = POS_{P}^{U}(D) \cup \{x\}$. Because $POS_{C}^{U'}(D) = POS_{C}^{U}(D) \cup \{x\}$, and P is the original feature subset, there is $POS_{P}^{U}(D) = POS_{C}^{U}(D)$, so $POS_{P}^{U'}(D) = POS_{C}^{U'}(D)$ holds. By Definition 1, it is necessary to further prove that $POS_{P'}^{U'}(D) \neq POS_{P}^{U'}(D)$ for any $P' \subset P$, similarly, this proof is similar as situation (1) of Proposition 3. Therefore, P is also the new feature subset of the changed incomplete decision system. (2) Since $S_P(x) \nsubseteq D_k(1 \leq k \leq r)$, which means there exists at least one object $x_i(1 \leq i \leq m)$ form tolerance relation with the new object x (otherwise $S_P(x) \subseteq D_k$). Because the tolerance relation is symmetric, there is $S'_P(x_i) = S_P(x_i) \cup \{x\}(1 \leq i \leq m)$, it also follows that $S'_P(x_i) \nsubseteq D_k(1 \leq k \leq n)$, where $S'_P(x_i)$ is a changed tolerance class. Clearly, all of the objects with changed tolerance classes are in $S_P(x)$, this means that it holds that $POS_P^{U'}(D) = POS_P^{U'}(D) - S_P(x)$. Because P is the original feature subset, it holds that $POS_P^U(D) = POS_C^U(D)$, and there is $POS_C^{U'}(D) = POS_C^U(D) \cup \{x\}$, thus it follows that $POS_P^{U'}(D) \neq POS_C^{U'}(D)$. Therefore, P is not the new feature subset of changed incomplete decision system. □

By the same way, for situation (1), we do not need to update the original compact discernibility matrix. However, for situation (2), we need to update the original matrix in an incremental manner. The new compact discernibility matrix M' is constructed as follows: for each original sub-discernibility matrix, the new object x is added to the decision classes with the same decision value, and modify the corresponding row; for other matrix elements, we keep them unchanged.

Case 3. Only forming a new decision class

In this case, $U'/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S'_C(x_i), \ldots, S_C(x_m), S_C(x)\}$, where for any changed tolerance class $S'_C(x_i)(1 \leq i \leq m)$, if $x \in S_C(x_i)$, then $S'_C(x_i) = S_C(x_i) \cup \{x\}$, and because the tolerance relation is symmetric, there is $S_C(x) = S_C(x) \cup \{x_i\}$. In addition, $U'/IND(D) = \{D_1, D_2, \ldots, D_r, D_{r+1}\}$, where $D_{r+1} = \{x\}$. Obviously, both $S_C(x) \nsubseteq D_{r+1}$ and $S'_C(x_i) \nsubseteq D_k(1 \leq k \leq r+1)$ holds. Clearly, all of the objects with changed tolerance classes are in $S_C(x)$, thus $POS_C^{U'}(D) = POS_C^U(D) - S_C(x)$.

Proposition 5. *If $|S_C(x)| > 1 \wedge |D_{r+1}| = 1$, P is also the new feature subset of the changed incomplete decision system.*

Proof. Since $S_C(x) \nsubseteq D_{r+1}$, $S'_C(x_i) \nsubseteq D_k(1 \leq i \leq m, 1 \leq k \leq r+1)$, and $P \subseteq C$, according to the partial relation, both $S_P(x) \nsubseteq D_{r+1}$ and $S_P(x_i) \nsubseteq D_k$ hold when a new object x immigrates into the system. Clearly, all these objects are in the new tolerance class $S_P(x)$, it follows that $POS_P^{U'}(D) = POS_P^U(D) - S_P(x)$. In addition, because P is the original feature subset, it follows that $POS_P^U(D) = POS_C^U(D)$, and there is $POS_C^{U'}(D) = POS_C^U(D) - S_C(x)$, thus it holds that $POS_P^{U'}(D) = POS_C^{U'}(D)$. By Definition 1, it is necessary only to prove that $POS_{P'}^{U'}(D) \neq POS_P^{U'}(D)$ for any $P' \subset P$. Since P is the original feature subset, it follows that $POS_{P'}^U(D) \neq POS_P^U(D)$ for any $P' \subset P$, by the partial relation, there is $POS_{P'}^U(D) \subset POS_P^U(D)$, thus it follows that $(POS_{P'}^U(D) - S_C(x)) \subset (POS_P^U(D) - S_C(x))$. Because there are $POS_{P'}^U(D) - S_C(x) = POS_{P'}^{U'}(D)$ and $POS_P^U(D) - S_C(x) = POS_P^{U'}(D)$, we can easily obtain that $POS_{P'}^{U'}(D) \subset POS_P^{U'}(D)$, thus $POS_{P'}^{U'}(D) \neq POS_P^{U'}(D)$. Therefore, P is also the new feature subset of the changed incomplete decision system. □

Case 4. Neither generating a new tolerance class nor a new decision class

In this case, $U'/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S'_C(x_i), \ldots, S_C(x_m), S_C(x)\}$, for any changed tolerance class $S_C(x'_i)(1 \leq i \leq m)$, if $x \in S_C(x_i)$, then $S'_C(x_i) = S_C(x_i) \cup \{x\}$, and because the tolerance relation is symmetric, there is $S_C(x) = S_C(x) \cup \{x_i\}$.

In addition, $U'/IND(D) = \{D_1, D_2, \ldots, D'_j, \ldots, D_r\}$, where $D'_j = D_j \cup \{x\}(1 \leq j \leq r)$. Here two situations may occur: a new object x is consistent with the set of original objects; a new object x is inconsistent with the set of original objects. For the first situation, it follows that $POS_C^{U'}(D) = POS_C^U(D) \cup \{x\}$. For the second situation, it follows that $POS_C^{U'}(D) = POS_C^U(D) - \{x_p \in S_C(x) || S'_C(x_p)/D| \neq 1\}$.

Proposition 6. *If $|S_C(x)| > 1$, P is also the new feature subset of the changed incomplete decision system.*

Proof. If a new object x is consistent with the set of original objects, then x forms tolerance relation with more than one objects. Because P is the original feature subset, it holds that $POS_P^U(D) = POS_C^U(D)$. In addition, since $P \subseteq C$, there is $POS_P^{U'}(D) = POS_P^U(D) \cup \{x\}$, thus it follows that $POS_P^{U'}(D) = POS_C^{U'}(D)$. By Definition 1, it is necessary only to prove that $POS_{P'}^{U'}(D) \neq POS_P^{U'}(D)$ for any $P' \subset P$, similarly, this proof is the same as situation (1) in Proposition 3. If a new object x is inconsistent with the original objects, since $P \subseteq C$, by the partial relation, it holds that $POS_P^{U'}(D) = POS_P^U(D) - S_C(x)$, thus there is $POS_P^{U'}(D) = POS_C^{U'}(D)$. It is necessary only to prove that $POS_{P'}^{U'}(D) \neq POS_P^{U'}(D)$ for any $P' \subset P$, similarly, this proof is the same as the proof in Proposition 5. As all analyzed above, P is also the new feature subset of the changed incomplete decision system. □

4.2. An efficient updated feature selection algorithm in dynamic incomplete decision systems

Based on the above considerations, we develop an efficient updated feature selection algorithm to compute a new feature subset in a dynamically-increasing incomplete decision system as follows.

Algorithm 1 An efficient feature selection algorithm based on the updated compact discernibility matrix (Algorithm FSUM)

```
Input: An incomplete decision system $IDS = (U, C \cup D, V, f)$, the original feature subset $Red$ and discernibility matrix $M$ , a new object $x$ ;
Output: A new feature subset $Red'$;
Begin
 1. Initialize $P \leftarrow Red$, $U' \leftarrow U \cup \{x\}$;
 2. Compute the tolerance classes $U/TR(C) = \{S_C(x_1), S_C(x_2), \ldots, S_C(x_m)\}$ and the decision classes $U/IND(D) = \{D_1, D_2, \ldots, D_r\}$;
 3. If $x \notin S_C(x_i)(1 \leq i \leq m)$ then
 4.     if $x \notin D_j(1 \leq j \leq r)$ and $|S_P(x)| = 1$ then turn to Step 21; // According to Proposition 3
 5.     else turn to Step 10;
 6.     if $x \in D_j(1 \leq j \leq r)$ and $S_P(x) \subseteq D_k$, then turn to Step 21; // According to Proposition 4
 7.     else turn to Step 10;
 8. Else //$x \in S_C(x_i)$
 9.     turn to Step 21; // According to Proposition 5 and Proposition 6
10. Calculate the updated discernibility matrix $M'$ incrementally; //According to discernibility matrix $M$
11. For $\forall p \in P$, if $\exists p \in m'(v, w)$ then //$m'(v, w)$ is a new matrix element in $M'$
12.     delete all the new matrix elements that contain feature $p$ ;
13. If $(c \in m'(v, w) \wedge |c| = 1)$ then
14.     let $P = P \cup \{c\}$, and delete all the new matrix elements that contain feature $c$; // According to Propositions 1 and 2
15. While $(M' \neq \phi)$ do
16.     $\forall b \in C - P$, let $b.count$=0; // $b.count$ denotes the frequency of feature $b$
17.     calculate the frequency of each feature $b.count$;
18.     select $b^{'} = max_{b \in C-P}\{b.count\}$, and let $P = P \cup \{b^{'}\}$;
19.     delete all the new matrix elements that contain feature $b^{'}$;
20. End while
21. $Red^{'} \leftarrow P$, return $Red^{'}$.
End
```

The main framework of Algorithm FSUM is as follows: Step 2 is to compute the tolerance classes and decision classes, the time complexity is $O(|U|^2|C|)$; Step 3-9 are to judge a new object x belongs to which case, the time complexity is $O(|U||C|)$; Step 10 is to update the discernibility matrix, the time complexity is $O(|U||C|)$; Step 11-12 are to delete the new matrix elements contained the features in P, the time complexity is $O(|U||C||P|)$; Step 13-14 are to add new singleton feature into the original feature subset by only scanning the new generated matrix elements, the time complexity is $O(|U|)$; Step 15-20 are to add a feature of the highest frequency from the existing features into the original feature subset, the time complexity is $O(|U||C-P|^2)$. This procedure is repeated until each new generated matrix element in M' is empty. Therefore, the total time complexity of Algorithm FSUM is $O(|U|^2|C|)$.

To stress above findings, we compare the time complexities of feature selection algorithms to find a new feature subset in a dynamic incomplete decision system shown as Table 1. For convenience, we denote the classical discernibility matrix-based feature selection algorithms from [2, 6, 24] to retrain such system as new one to compute a new feature subset as Algorithm FSCM, and the lower approximation-based feature selection algorithm of greedy and forward [28, 35] as Algorithm FSLA. Note that $|U|$ and $|C|$ denote the cardinalities of objects and condition features in the incomplete decision system, respectively.

Table 1. Comparison of complexities description

Algorithms	Computing a new feature subset
FSUM	$O(\|U\|^2\|C\|)$
FSCM	$O(\|U\|^2\|C\|^2)$
FSLA	$O(\|U\|^2\|C\|^3)$

From Table 1, we can see that the time complexity of Algorithm FSUM is lower than that of Algorithms FSCM and FSLA for feature selection, especial for microarray gene expression incomplete data sets, the number of gene features is usually larger than the number of objects, i.e., $|C| \gg |U|$. For this comparison, it is obvious that Algorithm FSUM is more efficient to find a new feature subset in dynamic incomplete decision systems.

From the aspect of storage spaces between the compact discernibility matrix and the classical discernibility matrix, the compact discernibility matrix only captures the non-empty matrix elements rather than all elements, thus the storage space of constructing the compact matrix is $O(|C|(|U|^2-\sum_{i=1}^{r}|D_i|^2))$. However, the classical discernibility matrix is constructed as follows: the total numbers of rows and columns are both $|U|$, each matrix element $m(i,j)$ of the matrix corresponding to objects x_i and x_j includes the conditional features in which the feature values of two objects can discern. Thus the storage space of constructing the classical matrix is $O(|C||U|^2)$. Therefore, using the compact discernibility matrix, the search space is greatly compressed and the computational load is reduced in feature selection tasks, especially for the incomplete decision systems with imbalance decision classes.

5. EXPERIMENTAL EVALUATIONS

The objective of the following experiments is to show the efficiency and effectiveness of the proposed feature selection algorithm for selecting a feature subset. Data sets used in the experiments are outlined in Table 2, which are downloaded from UCI Machine Learning Repository [29]. For the data sets with missing values, the missing values of which are represented by the set of all possible values of each attribute. For the artificial dataset, the feature values of which are generated randomly from 0 to 9. All the experiments are conducted on a PC with Windows 7, Intel (R) Core(TM) Duo CPU 2.93 GHz and 4GB memory. Algorithms are coded in C++ and the software being used is Microsoft Visual 2008.

In the following, there are three objectives to conduct the experiments. One is to show the feasibility of compact discernibility matrix. The other is to test the computational efficiency of the proposed feature selection algorithm to find a new feature subset. Another is to evaluate the classification performance of selected feature subsets from feature selection phase.

Table 2: The detailed information of the data sets used in the comparison

ID	Data sets	Samples	Features	Classes
1	Lung Cancer	32	56	3
2	Hepatitis	155	19	2
3	Soybean-large	307	35	19
4	Voting Records	435	16	2
5	Breast Cancer Wisconsin	699	10	2
6	Mushroom	8124	22	2
7	Aritificial Dataset	1000	5000	10

5.1. Feasibility of compact discernibility matrix

In this subsection, we will show the feasibility of the proposed compact discernibility matrix in feature selection tasks. For each data set shown in Table 2, Algorithms FSCM, FSUM and FSLA mentioned in Section 4.2 are employed to find new feature subsets, respectively. A comparative study in terms of feature subset size and running time is executed on the seven data sets. Note that the main difference between Algorithms FSCM and FSUM is the construction of discernibility matrix. Table 3 compares the subset size and running time for Algorithms FSCM, FSUM and FSLA.

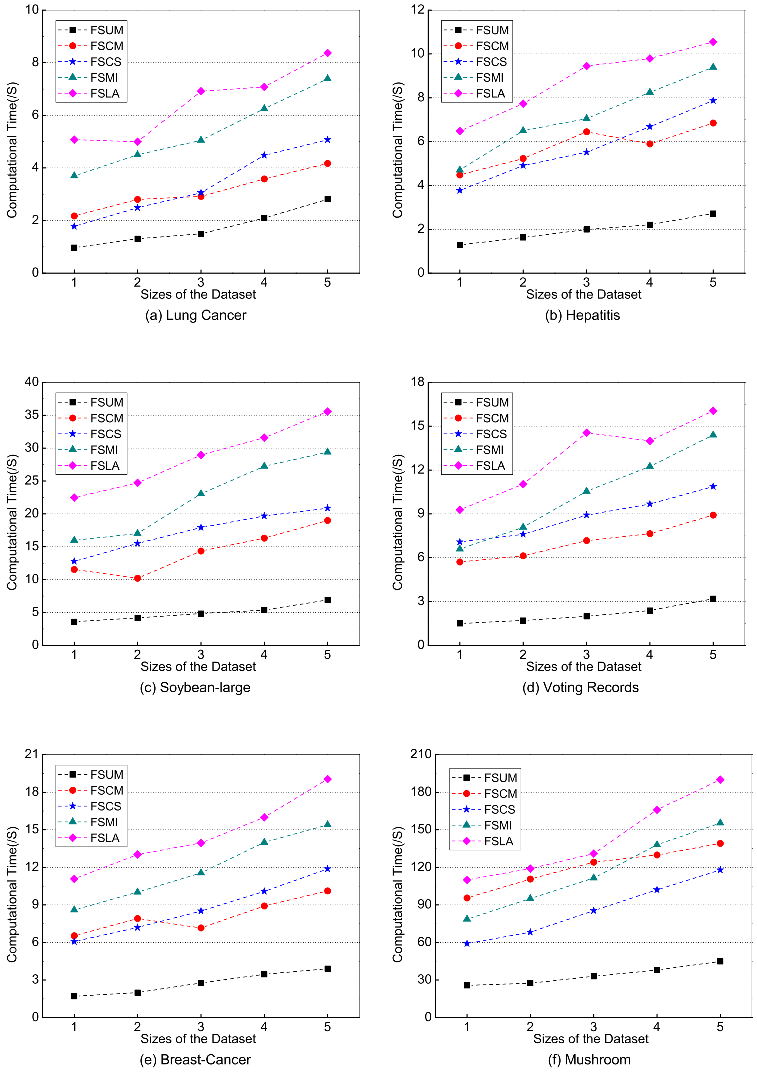
Table 3. Comparison of feature subset size and running time

	Feature Subset Size			Running Time(s)		
ID	FSUM	FSLA	FSCM	FSUM	FSLA	FSCM
1	4	4	4	4.315	9.506	7.248
2	7	7	7	6.487	21.242	15.183
3	13	12	13	17.520	40.038	28.915
4	9	9	9	7.729	18.641	13.704
5	4	4	4	9.510	20.278	18.133
6	5	4	5	131.634	390.575	284.370
7	18	17	18	192.953	807.406	515.368

From Table 3, it is easy to see that all the results of feature subset size by the three feature selection algorithms are similar or equal, which verifies the validity of the compact discernibility matrix in feature selection tasks. In addition, it is clear from the running time figures that Algorithms FSUM and FSCM run considerably faster than FSLA, the main reason is that FSUM and FSCM do not need to compute the time-consuming lower approximation in the data sets. However, the running time of Algorithm FSUM is smaller than that of FSCM. This primarily, can be attributed to the data sets with imbalance decision classes such that much time is not taken in calculating the pairwise-comparison of objects that belong to the same decision classes. For data set Hepatitis, the number of the first decision class is 32, while that of the second decision class is 123, and for data set Breast Cancer Wisconsin, the number of the first decision class is 458, while that of the second decision class is 241. The imbalance of the decision classes in the two data sets is obvious. The comparative results between FSUM and FSCM in terms of running time demonstrate that the discernibility matrix in feature selection tasks is indeed compact, such that the running time is saved. Therefore, we can get the conclusion that the compact discernibility matrix is feasible from the experimental results.

5.2. Efficiency of proposed feature selection algorithm

To illustrate the computational efficiency of proposed feature selection algorithm for selecting a new feature subset from a dynamically-increasing incomplete decision system. In what follows, FSUM is experimentally evaluated with four algorithms, two are FSCM and FSLA respectively, and other two leading feature selection methods, statistical approach chi-square [37] (denoted by FSCS) and information theoretic approach mutual information [38, 39] (denoted by FSMI). Since the number of features selected by FSCS and FSMI is different, for the sake of impartiality, we select the same quantity of features and the selected features are ordered in a descending sequence by their priorities. To distinguish the running time, for each data set shown in Table 2, 75% objects are taken as the original data set, and 25% objects are divided into five equal parts of equal size. The first part is regarded as the first immigrational object set, the combination of the first immigrational object set and the second part is viewed as the second immigrational object set, ..., the combination of all five parts is viewed as the fifth immigrational object set. In the following figure, the *x*-coordinate pertains to the size of data set, while the *y*-coordinate concerns the running time, the running time is expressed in seconds.



(a) Lung Cancer

(b) Hepatitis

(c) Soybean-large

(d) Voting Records

(e) Breast-Cancer

(f) Mushroom

Fig.1. Comparison of running time among different feature selection algorithms

It can be observed from Fig.1 that the larger size of immigrational object set is, the longer running time of the five algorithms is. But the curves are not strictly monotonic for FSCM and FSLA, specifically, the curves fluctuate significantly. Take FSCM as an illustration, as the size of data set from 85% to 90% in data set Hepatitis, the running time decreases. As to some other data sets, the same phenomenon can be seen. The main reason is that the number of selected features is different in the data sets is different. Clearly, we can see from each data set that the running time of FSLA and FSMI approach is longer than that of other methods, the reason can be explained by the fact that the running time of lower approximation in FSLA is relatively expensive, and the computation of mutual information in FSML costs much time. One can also observe that the curve corresponds to FSUM increases the most slowly among the five approaches when an immigrational object set adds to the data sets, furthermore, the effect is more obvious for large-scale incomplete data sets. This phenomenon coincides with the incremental manner on feature selection, FSUM avoids some recalculations, making use of previous computational results, such that the time efficiency of computing new feature subset is improved. And the curve corresponds to FSUM increases slowly in some data sets, such as data set Mushroom, as the size of data set from 80% to 85%, the running time is nearly the same. The reason is explained as the immigrational object set may result in the lower approximation unchanged, such that the original selected feature subset remains constant. Consequently, the time efficiency is improved. The phenomenon indicates that the variation of lower approximation and incremental manner have great influence on feature selection in dynamic incomplete data sets. From the above experimental results, we can affirm that the proposed feature selection algorithm outperforms four state-of-the-art algorithms in dynamic incomplete decision systems.

To further demonstrate the time efficiency of FSUM on large-scale data set where the number of features is significantly greater than the number of samples, for the Artificial Dataset, we select 500 objects as the original data set, the remaining objects are divided into ten parts of equal size, each time 50 objects are added to the data set. We compare the running time of the five feature selection algorithm with the size of data set from 500 to 1000. Fig.2 displays the detailed trend of the five algorithms in terms of running time as the size of the data set.

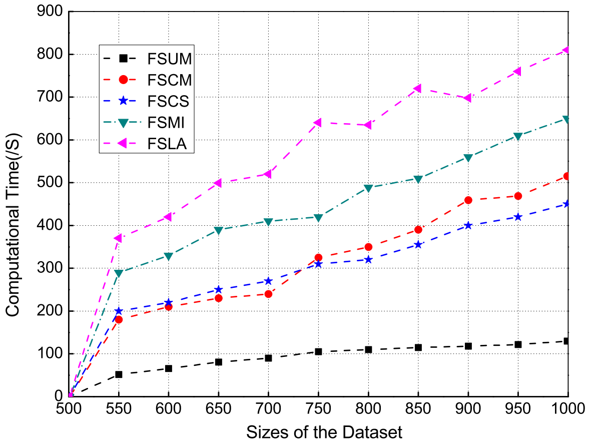


Fig.2. Running time of different algorithms on the Artificial Dataset

From Fig.2, we can see that FSUM spends less time than other four feature selection methods on the Artificial Dataset. Clearly, FSUM exhibits the best time efficiency on this data set and FSLA performs the worst. The running times of FSCS and FSCM are close, but the former is slightly lower than the latter. The experimental result indicates that FSUM has stronger competitiveness in feature selection from dynamic large-scale data sets with numbers of features in terms of time efficiency.

5.3. Comparative analysis of classification performance

In the sequence of experiments, two classifiers are employed to evaluate the classification performance of selected feature subsets from feature selection phase: C4.5 and Support Vector Machine (SVM). After feature selection by above various feature selection methods, the data sets are reduced according to the selected feature subsets. The reduced datasets are then classified using the two classifiers. As to SVM, the pre-processing step includes encoding all categorical features into binary attributes and scaling each numerical feature to be in the range [0, 1]. Tables 4 and 5 display the average classification accuracy as a percentage obtained using 10 cross-validation. Because the main difference between FSUM and FSCM is the time efficiency in feature selection, they select the same feature subsets, which keep the same classification performance with each other. Thus here we no longer focus on the classification accuracies of FSCM.

Table 4. C4.5 performance of the subsets of selected features with different feature selection algorithms

		Classifier C4.5			
ID	Data sets	FSUM	FSLA	FSCS	FSMI
1	Lung Cancer	86.15±1.02	84.07±1.56	85.42±0.95	80.73±1.13
2	Hepatitis	81.35±0.67	79.55±0.91	82.18±0.50	79.07±0.84
3	Soybean-large	90.26±1.43	87.91±1.68	89.39±1.61	85.42±1.75
4	Voting Records	93.85±1.10	90.33±1.05	91.95±1.32	90.54±1.49
5	Breast Cancer Wisconsin	78.54±0.38	76.02±0.59	79.11±0.78	74.21±0.51
6	Mushroom	100.0±0.00	100.0±0.00	100.0±0.00	100.0±0.00
7	Aritificial Dataset	97.45±0.28	95.96±0.31	97.02±0.51	96.84±0.65

Table 5. SVM performance of the subsets of selected features with different feature selection algorithms

		Classifier SVM			
ID	Data sets	FSUM	FSLA	FSCS	FSMI
1	Lung Cancer	89.13±0.76	85.25±0.30	86.94±0.85	82.71±0.55
2	Hepatitis	82.99±1.09	81.02±1.26	82.97±1.13	80.65±1.41
3	Soybean-large	93.28±0.50	88.57±0.33	94.15±0.69	89.03±0.72
4	Voting Records	92.76±0.91	86.38±0.54	87.05±1.10	90.89±0.78
5	Breast Cancer Wisconsin	85.15±1.64	82.66±2.01	84.32±1.95	77.90±1.83
6	Mushroom	100.0±0.00	100.0±0.00	100.0±0.00	100.0±0.00
7	Aritificial Dataset	99.43±0.15	99.78±0.47	99.01±0.42	97.95±0.80

The comparison results from Table 4 show that for the C4.5 classifier, the feature subset selected by FSUM obtains the highest classification accuracy four times, while FSCS attains the highest accuracy two times. It also can be observed from Table 5 that FSUM exhibits the highest classification accuracy of the SVM classifier in most of the data sets. Out of seven data sets, FSUM achieves highest classification accuracy in four data sets, while both FSLA and FSCS attain highest classification accuracy in one data set. However, the classification accuracies obtained by the SVM classifier are higher than those obtained by the C4.5 classifier. This experimental result can be explained in that the selected feature subsets from each data set are well suited to the SVM classifier. From the results shown in Tables 4 and 5, we can draw a conclusion that the classification performance of FSUM is better than that of other feature selection algorithms in most of the data sets irrespective of classifiers used. The better performance of FSUM is achieved due to the fact that FSUM can select relevant and significant features from dynamic incomplete data more efficiently than other feature selection algorithms. From the standard deviations reported in Tables 4 and 5, it also can be seen that FSUM has a better robustness than other feature selection algorithms in most of the data sets.

All the experimental results mentioned above demonstrate the efficiency and effectiveness of the proposed feature selection algorithm, which provides a useful approach for feature selection from dynamic incomplete decision systems.

6. CONCLUSIONS

Research on feature selection in dynamic incomplete decision systems has shown its importance in real-life applications. In this paper, we have developed an efficient updated feature selection algorithm based on the proposed compact discernibility matrix. In the compact discernibility matrix, only non-empty matrix elements are captured, such that the computational time is saved to complete the feature selection task in a dynamically-increasing incomplete decision system. The experimental results pertaining to different data sets have substantiated the proposed algorithm is effective and efficient in dynamic incomplete decision systems. In future work, we will focus on how to calculate a new feature subset when an object set in other complex incomplete decision systems evolves over time, such as dominance information decision systems, incomplete information systems with fuzzy decision.

REFERENCES

1. M. Kryszkiewicz, “Rough set approach to incomplete information systems,” *Information Sciences*, vol.112, 1998, pp.39-49.

2. A. Skowron and C. Rauszer, “The discernibility matrices and functions in information systems,” in:R.Slowinski(Ed.). *Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer, Dordrecht, 1992.

3. Q.H. Hu, Z.X. Xie and D.R. Yu, “Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation,” *Pattern Recognition*, vol.40, 2007, pp.3509-3521.

4. Y.H. Qian, J.Y. Liang and C.Y. Dang, “Incomplete multi-granulations rough set,” *IEEE Transactions on Systems, Man and Cybernetics: Part A*, vol.40, 2010, pp.420-431.

5. N.S. Jaddi and S. Abdullah, “Nonlinear great deluge algorithm for rough set attribute reduction”, *Journal of Information Science and Engineering,* vol.29, 2013, pp.49-62.

6. Z. Pawlak and A. Skowron, “Rough sets and Boolean reasoning”, *Information Sciences*, vol.177, 2007, pp.41-73.

7. Y.H. Qian, J.Y. Liang, W. Pedrycz and C.Y. Dang, “Positive approximation: An accelerator for attribute reduction in rough set theory,” *Artificial Intelligence*, vol.174, 2010, pp.597-618.

8. M. Kryszkiewicz and P. Lasek, “FUN: fast discovery of minimal sets of attributes functionally determining a decision attribute,” *Lecture Notes in Computer Science*, vol.4585, 2007, pp.320-331.

9. P. C. Wang, “Highly scalable rough set reducts generation,” *Journal of Information Science and Engineering*, vol.23, 2007, pp.1281-1298.

10. I. Kim, Y.Y. Chu, J.Z. Watada, J.Y. Wu and W. Pedrycz, “A DNA-based algorithm for minimizing decision rules: a rough sets approach,” *IEEE Transactions on Nanobioscience*, vol.10, 2011, pp.139-151.

11. M.L. Othman, I. Aris, S.M. Abdullah, M.L. Ali and M.R. Othman, “Knowledge discovery in distance relay event report: a comparative data-mining strategy of rough set theory with decision tree,” *IEEE Transactions on Power Delivery*, vol.25, 2010, pp.2264-2287.

12. Y. Leung and D.Y. Li, “Maximal consistent block technique for rule acquisition in incomplete information systems,” *Information Sciences*, vol.153, 2003, pp.85-106.

13. Y.T. Xu, L.S. Wang and R.Y. Zhang, “A dynamic attribute reduction algorithm based on 0-1 integer programming,” *Knowledge-Based Systems*, vol.24, 2011, pp.1341-1347.

14. J.W. Grzymala-Busse, “Data with missing attribute values: generalization of indiscernibility relation and rule reduction. Transactions on Rough Set I”, *Lecture Notes in Computer Science*, vol.3100, Spring-Verlag, Berlin, 2004.

15. Z.Q. Meng and Z.Z. Shi, “A fast approach to attribute reduction in incomplete decision systems with tolerance relation-based rough sets,” *Information Sciences*

16. D. Liu, T. Li, D. Ruan and W. Zou, “An incremental approach for inducing knowledge from dynamic information systems,” *Fundamenta Informaticae*, vol.94, 2009, pp.245-260.

17. M.Yang, “An incremental updating algorithm for attribute reduction based on improved discernibility matrix,” *Chinese Journal of Computers*, vol.30, 2007, pp.815-822.

18. G.Z. Luo and X.B. Yang, “Limited dominance-based rough set model and knowledge reductions in incomplete decision system,” *Journal of Information Science and Engineering,* vol.26, 2010, pp.2199-211.

19. Y.H. Qian, J.Y. Liang, D.Y. Li, F. Wang and N.N. Ma, “Approximation reduction in inconsistent incomplete decision tables,” *Knowledge-Based Systems,* vol.23, 2010, pp.427-433.

20. N.M. Parthalain, Q. Shen and R. Jensen, “A distance measure approach to exploring the rough set boundary region for attribute reduction,” *IEEE Transactions on Knowledge and Data Engineering*, vol.22, 2010, pp.305-317.

21. F. Hu, G.Y. Wang, H. Huang and Y. Wu, “Incremental attribute reduction based on elementary sets,” in *Proceeding of the 10th International Conference on Rough sets, Fuzzy sets, Data mining and Granular computing*, 2005, pp.185-193.

22. Y. Zhao, Y.Y. Yao and F. Luo, “Data analysis based on discernibility and indiscernibility,” *Information Sciences*, vol.177, 2007, pp.4959-4976.

23. K.M. Gupta, W.A. David and M. Philip, “Rough set feature selection algorithms for textual case-based classification,” in: T.R.Roth-Berghofer, et al.(Eds), *Lecture Notes in Artificial Intelligence* vol.4106, 2006, pp.166-181.

24. F.F. Xu, D.Q. Miao and L.Wei, “Fuzzy-rough attribute reduction via mutual information with an application to cancer classification,” *Computers & Mathematics with Applications*, vol.57, 2009, pp.1010-1017.

25. D. Tian, X.J. Zeng and J. Keane, “Core-generating approximate minimum entropy discretization for rough set feature selection in pattern classification,” *International Journal of Approximate Reasoning*, vol.52, 2011, pp.863–880.

26. Y.H. Qian, J.Y. Liang and F. Wang, “A new method for measuring the uncertainty in incomplete information systems,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol.17, 2009, pp.855-880.

27. R. Kohavi and G.H. John, “Wrappers for feature subset selection,” *Artificial Intelligence*, vol.97, 1997, pp.273-324.

28. E.C.C. Tsang, D.G. Chen, D.S. Yeung, X.Z. Wang and J.W.T. Lee, "Attribute reduction using fuzzy rough sets," *IEEE Transactions on Fuzzy Systems*, vol.16, 2008, pp.1130-1141.

29. UCI Machine Learning Repository, http://archive.ics.uci.edu/ ml/datasets. html.

30. Y.H. Qian, J.Y. Liang and C.Y. Dang, "Converse approximation and rule extraction from decision tables in rough set theory," *Computers & Mathematics with Applications*, vol.55, 2008, pp.1754-1765.

31. M. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining," *IEEE Transactions on Knowledge and Data Engineering*, vol.15, 2003, pp.1437-1447.

32. L. Sun, J.C. Xu and Y. Tian, "Feature selection using rough entropy-based uncertainty measures in incomplete decision systems," *Knowledge-Based Systems*, vol.36, 2012, pp.206-216.

33. R. Jensen and Q. Shen, "Fuzzy-rough sets assisted attribute selection," *IEEE Transactions on Fuzzy Systems*, vol.15, 2007, pp.73-89.

34. W.Z. Wu, M. Zhang, H.Z. Li, and J.S. Mi, "Knowledge reduction in random information systems via Dempster–Shafer theory of evidence," *Information Sciences*, vol.174, 2005, pp.143-164.

35. T.Li, and D.Ruan, "A rough sets based characteristic relation approach for dynamic attribute generalization in data mining," *Knowledge-Based Systems*, vol.20, 2007, pp.485-494.

36. S. Das, "Filters, wrappers and a boosting-based hybrid for feature selection," in *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001, pp.74-81.

37. X. Jin, A. Xu, R. Bie and P. Guo, "Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles," *Lecture Notes in Computer Science*, vol.3916, 2006, pp.106-115.

38. D.Huang and T.W.S.Chow, "Effective feature selection scheme using mutual information", *Neurocomputing*, vol.63, 2004, pp.325-343.

39. R.Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on Neural Network*, vol.5, 1994, pp.537-550.

40. W.H.Shu and H. Shen, “Updating attribute reduction in incomplete decision systems with the variation of attribute set ,” *International Journal of Approximate Reasoning,* 2013, In Press.

41. F.Wang, J.Y. Liang, et al., “Attribute reduction for dynamic data sets,” *Applied Soft Computing*, vol.13, 2013, pp. 676-689.