Knowledge-Based Systems 279 (2023) 110947

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Matrix-based feature selection approach using conditional entropy for ordered data set with time-evolving features

Weihua Xu *, Yifei Yang

College of Artificial Intelligence, Southwest University, Chongqing, 400715, PR China

ARTICLE INFO

Article history:
Received 17 January 2023
Received in revised form 29 May 2023
Accepted 28 August 2023
Available online 1 September 2023

Keywords:
Conditional entropy
Feature selection
Incremental algorithm
Ordered data set
Rough set

ABSTRACT

With the successful application of rough sets in many fields, research results on the theory emerge one after another. As one of the core contents of rough set theory, feature selection aims to find the minimum attribute set that does not affect the overall classification ability. In real life, there are often some data whose features will change with variables such as time, which is called ordered data with changing features. However, for ordered data with changing features, the existing methods in the current field are not applicable, because when the features of the data change, these methods basically need to recalculate from scratch to get a new reduction result, which is very time-consuming and does not use the previous reduction result. The feature incremental attribute reduction algorithm can be applied to the previous reduction results, thus greatly saving time. Drawing inspiration from this, this paper studies incremental attribute reduction algorithms under the order data with changing features. This paper first gives the entropy of the dominant condition of the dominant relation matrix and the updating principle of the new dominant relation matrix and the dominant diagonal matrix when the feature changes are explored. Later, two incremental attribute reduction algorithms HAR-A and HAR-D are also proposed in this paper, which are respectively applied to add features and delete features in ordered data with changing features. The subsequent experiments were also carried out on 9 data sets of UCI, and the performance of the proposed algorithm was evaluated. It can be seen from the experimental results that the two incremental attribute reduction algorithms we proposed are very effective.

1. Introduction

Attribute reduction, as the core problem of rough set theory research, not only can effectively reduce the dimension of data, but also the reduction results have clear semantic interpretation, so it has attracted wide attention [1–4]. The so-called attribute reduction refers to the deletion of redundant attributes in data by using constraints constructed on certain metrics to improve the performance of subsequent learning algorithms. In real life, data sets often change. For example, the characteristics of data sets change with time and other variables, which is called dynamic data sets. The attribute reduction algorithm for dynamic data sets is generally incremental attribute reduction method [5–9]. Incremental attribute reduction method can effectively use the existing reduction results, thus saving a lot of time and space costs, so it has attracted much attention. Based on this problem, an feature incremental attribute reduction method for dynamic ordered data set features is studied in this paper.

With the development of the information age, the complexity and diversity of data structures are increasing, and the feature selection methods are constantly improved and innovated. Many excellent feature selection models and algorithms were proposed already. Some usually used feature representation methods based on deep learning are convolutional neural network (CNN) [10], Restricted Boltzmann machine (RBM) [11] and recursive neural network (RNN) [12]. Recently, deep learning has also been used in some problems about attribute reduction. Zhao's team described a feature selection algorithm based on multi-dimensional DNN and relatively rare population ring [13]. Semwal's team described a very robust feature extraction method and applied it to classification problems [14]. Chen's team has completed a method of target feature prediction based on electroencephalogram [15]. In addition, in real life problems, this target feature prediction method has been successfully applied in the fields of economy [16], remote control [17], transportation [18] and so on. Evolutionary algorithm is inspired from Darwin's evolution theory. As the name implies, it carries out feature selection and optimization problems through behaviors similar to the evolution of various organisms in nature [19]. The Nag's team used multi-objective genetic programming to study feature extraction and selection methods of simplified classifiers [20]. Labani's

* Corresponding author.
E-mail addresses: chxuwh@gmail.com (W.H. Xu), Yifei.Yang.yyf2002@gmail.com (Y.F. Yang).

https://doi.org/10.1016/j.knosys.2023.110947
0950-7051/

team proposed and proved a multi-objective genetic algorithm for text feature selection based on relative criteria [21]. Ma's team proposed a feature selection method on the basis of genetic programming for classification [22]. Das's team described a feature extraction algorithm by simultaneously studying two targets [23]. Li's team demonstrated a method to simultaneously study the characteristics of multiple targets and combine it with genetic algorithm [24].

Rough set theory is a very significant theoretical basis for attribute reduction problems [25–27]. Pawlak proposed that RST is an effective mathematical tool to deal with inconsistent and uncertain information [28]. Later, with the advantages of rough set theory in processing imprecise and incomplete data, it is gradually known by the majority of scholars. As the core content of rough set, attribute reduction is to gradually remove unnecessary attributes from the attribute set to save much time from the data processing. At present, many scholars have proposed their own improved attribute reduction algorithms from the aspects of knowledge division, closeness, mutual information, granularity, etc.Because rough set theory does not meet the standard inconsistent requirement that exist in terms of credit ratings, article rankings, and profit margins with properties that have preferred ranking domains. Due to this shortcoming, Greco's team described a feature extraction method based on dominance relationship [29], and this proposed method has also achieved success in multiple dimension prediction and decision making [30]. Rough set method of monotone variable consistency [31], rough set model on the basis of random dominance [32], rough set model on the basis of soft dominance [33] and a rough set model that we often describe [34]. The above models and methods are very enlightening to this paper.

In real life, the characteristics of ordered data often change with variables such as time and age. Such as, a person's height, a student's grades. Some of the data is dynamic as students graduate and enter school. For dynamically ordered data sets, calculating reductions using these existing methods is very time consuming because they require recalculating knowledge, no one is born to know everything, the accumulation of knowledge is a long process, the accumulation of knowledge slowly from zero to full, from nothing to something. In this case, if the calculation of attribute reduction is carried out from the beginning every time, it will consume a lot of time and space, which is not worthwhile, so we need a dynamic incremental attribute reduction algorithm.

Dynamic attribute reduction algorithms can be roughly divided into three types: one is to change the data set sample algorithm, one is to change the data set attribute characteristics algorithm, and one is to change the data set attribute eigenvalue algorithm.

For changing the data set sample. Liang's team described a dynamic update algorithm on the basis of information values [35]. Zhang and his team demonstrated a dynamic selection algorithm for actively selecting sample features [36]. Yang's team studied dynamics centered on attribute characteristics method based on the active sample selection principle [37], then feature extraction was carried out for the isomeric data that would change [38]. Shu team described a dynamic feature extraction algorithm that crushes various kinds of data together [39]. Ye team proposed and proved a dynamic algorithm related to matrix pseudo-values [40]. Das team demonstrated a grouping dynamic algorithm combined with genetic algorithm [41].

For changes in attributes. Chen team introduced an incremental attribute reduction method on the basis of identifiable relation to dynamically increase attributes [42]. Wang's team conducted an algorithm related to information entropy for data sets subject to dynamic changes [43]. The Lang team proposed and proved a family-related algorithm [44]. Zeng Team studied an incremental attribute reduction method for mixed data on the basis of fuzzy rough sets [45].

The alternations of the attributes of the dataset. Wang's team described an algorithm based on representative entropy values [46]. Wei's team demonstrated a feature selection method based on discrimination matrix [47], and then developed an accelerated increment algorithm based on the compressed decision table technique [48]. The Cai team proposed and proved a coarse-grained dynamic attribute reduction algorithm and a fine-grained dynamic attribute reduction algorithm [49]. On this basis, Dong and Chen proposed a new incremental attribute reduction algorithm based on RST for the decision table with both samples and attributes increasing at the same time [50]. Jing team introduced an incremental method for calculating the reduction of decision tables with both objects and attributes evolving over time [51].

Through the study of the above algorithms, it is found that the algorithms that change the attribute characteristics of the data set are not based on the ordered data set. Therefore, it is urgent to put forward an algorithm to change the attributes of ordered data sets, which also inspires the author of this paper.

As a measure of uncertainty, information entropy has attracted wide attention. After Shannon [52] proposed information entropy, relevant researches have been extended. For data with sequential relationship, Hu's team introduced the basic concepts of ascending conditional entropy and descending conditional entropy [53]. In the following content of this paper, ascending and descending conditional entropy will also be used to pave the way for our proposed attribute reduction method.

Since information in matrix form the calculation process can be reduced to reduce the complexity of the algorithm, and the matrix related computing technology will be introduced into the dynamic algorithm. In addition, the correlation between objects in DRSA is an antisymmetric preference order relation. Therefore, what DRSA constitutes is an irregular space. As a result, it would be tedious and complex to use collection represents a kind of question about how to study DRSA, and it is really worth going into, This is especially true for non-static and sequential data sets. Therefore, a simple and effective method is needed to cover-matrix method based approximate spatial knowledge acquisition method, where situations are dynamic and require efficient knowledge acquisition. For the dominance matrix relationship, this paper studies the increment mechanism of the dominant conditional entropy by using matrix form.

After the above description, we find that dynamic attribute reduction by changing attribute characteristics is a worthy research direction. Furthermore, for the sake of improving the efficiency of the algorithm, unnecessary attributes in the alternative non-kernel set attribute set are gradually deleted in this paper when solving the reduction, and the original dominant relation matrix does not need to be repeatedly calculated. Therefore, this paper mainly studies the progressive method about DRSA based dynamic attribute reduction method based on non-static and sequential data sets. The main contribution of this paper are as follows : (1) A matrix-based method for calculating the dominant conditional entropy in ordered information systems is proposed and proved. (2) Two incremental feature attribute reduction algorithms HAR-A and HAR-D are proposed and demonstrated for changing attribute features of ordered data sets. They are applied to add and delete multiple attribute features, respectively. (3) Experiments on 9 data sets of UCI show that the proposed algorithm is effective.

The organization of this document is as follows. The second section presents the relevant basic knowledge. In Section 3, a computational method of conditional entropy of dominance based on dominant relation matrix calculation method is introduced and proved, and a heuristic attribute reduction algorithm

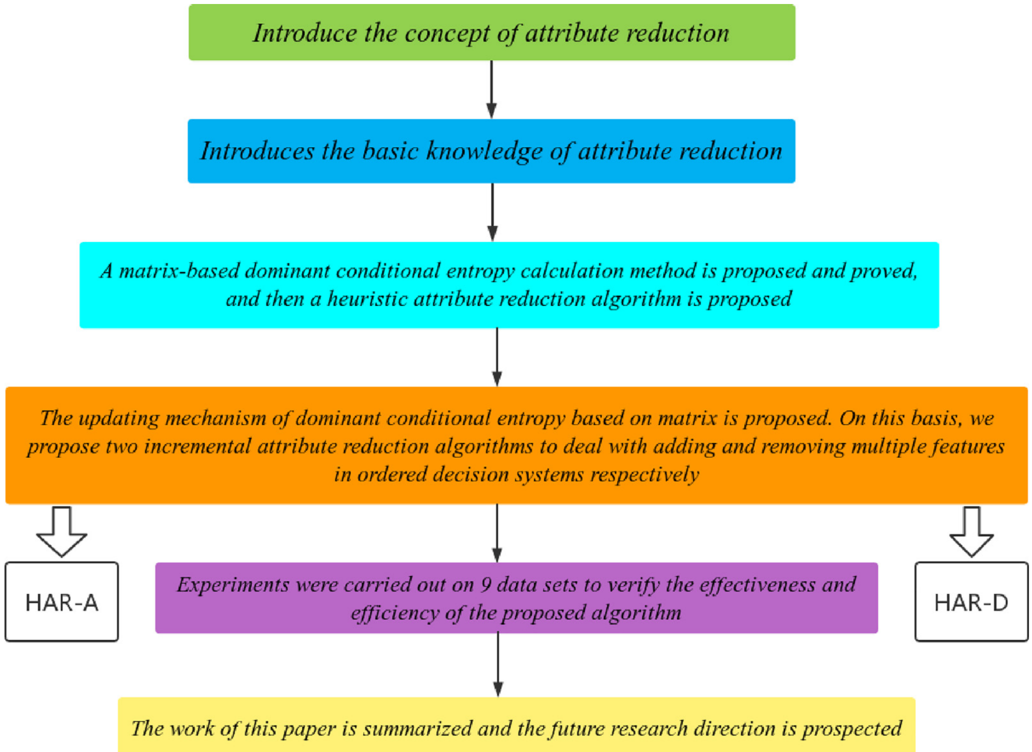


Fig. 1. Motivations of our work.

is proposed based on it. In Section 4, Introduce and update the inheritance mechanism of matrix dominant conditional entropy when the data set changes. Two incremental feature attribute reduction algorithms HAR-A and HAR-D are introduced to deal with adding and removing multiple features in ordered decision systems. In Section 5, experimental results on 9 data sets are presented to verify the effectiveness, efficiency and the performance evaluation of the proposed algorithm. Section 6 summarizes the work of this paper and looks forward to the future research direction. Motivations of our work is shown in Fig. 1.

2. Preliminaries

In this section, we describe some of the basics of DRSA.

2.1. The basics of attribute reduction

Definition 2.1 ([28]). Let an information system be denoted by a 4-tuple $S = (U, AT, V, f)$, where $U = \{x_1, x_2, \ldots, x_n\}$ is a non-empty finite set of objects; AT is a non-empty finite set of attributes; $V = \bigcup_{a\in AT} V_a$, V_a is the domain of attribute a; $f : U\times AT \to V$ is the information function with $f(x, a) \in V_a$, $\forall a \in AT$ and $x \in U$.

In an information system, an attribute is a criterion if its domain is sorted according to an ascending or descending preference. When all of its properties are criteria, it is called ordered information system(OIS) and expressed as $S^{\geq} = (U, AT, V, f)$.

In real life, the attribute eigenvalues of most ordered data are in order. For example, the higher the better when it comes to wages and stores' operating profits; For bankruptcy odds, all else being equal, the lower the better.

Definition 2.2 ([32]). Given $S^{\geq} = (U, AT, V, f)$ is an OIS, $\forall P \subseteq AT, P \neq \emptyset$, the conditional relation with ascending order D_P is like this

$$D_P = \{(x, y) \in U \times U : f(x, a) \geq f(y, a), \forall a \in P\}. \quad (1)$$

Table 1
A score table of etiquette assessment.

	a_1	a_2	a_3	a_4	d
x_1	98	95	96	99	A
x_2	87	88	86	89	B
x_3	78	75	74	78	C
x_4	66	65	67	63	D
x_5	54	53	43	37	E

Property 2.1 ([32]). *When D_P is a dominance relation in an ordered information systems, it has the following properties.*

(1) Reflexive: $\forall x \in U$, then xD_Px;

(2) Non-symmetric: $\forall x, y \in U$, let xD_Py, then yD_Px cannot be taken as true;

(3) Transitive: $\forall x, y, z \in U$, let xD_Py and yD_Pz, then xD_Pz.

Definition 2.3 ([32]). Given $S^{\succeq} = (U, AT, V, f)$ is an OIS, $\forall P \subseteq AT, P \neq \emptyset$, the two relational sets of x are called N-dominating sets and N-dominated sets, respectively, and they are defined like this

$$D_N^+(x) = \{y \in U : yD_Nx\}; \quad (2)$$

$$D_N^-(x) = \{y \in U : xD_Ny\}. \quad (3)$$

Example 1. Table 1 is a score table of etiquette assessment, where a_1, a_2, a_3, and a_4 represent etiquette 1, etiquette 2, etiquette 3, and etiquette 4 respectively, and x_1, x_2, x_3, x_4, and x_5 represent five people, where $P = \{a_1, a_2, a_3, a_4\}$, $U = \{x_1, x_2, x_3, x_4, x_5\}$, D_P is a dominance relation.

After observing Table 1, we take Table 1 as an example for the following proof. (1) $\forall x \in U$, then xD_Px holds; (2) $x_1D_Px_2$ holds, but $x_2D_Px_1$ does not hold; (3) $x_1D_Px_2$ and $x_2D_Px_3$ hold, then $x_1D_Px_3$ also holds. $D_P^+(x_1) = \{x_1\}$, $D_P^+(x_2) = \{x_1, x_2\}$,

$D_P^+(x_3) = \{x_1, x_2, x_3\}$, $D_P^+(x_4) = \{x_1, x_2, x_3, x_4\}$, and $D_P^+(x_5) = \{x_1, x_2, x_3, x_4, x_5\}$; $D_P^-(x_1) = \{x_1, x_2, x_3, x_4, x_5\}$, $D_P^-(x_2) = \{x_2, x_3, x_4, x_5\}$, $D_P^-(x_3) = \{x_3, x_4, x_5\}$, $D_P^-(x_4) = \{x_4, x_5\}$, and $D_P^-(x_5) = \{x_5\}$.

Property 2.2 ([32]). *For any $P_1, P_2 \subseteq AT$ and $\forall x \in U$, the following properties hold.*

(1) Let $P_1 \subseteq P_2$, then $D_{P_2}^+(x) \subseteq D_{P_1}^+(x)$ and $D_{P_2}^-(x) \subseteq D_{P_1}^-(x)$;

(2) $D_{P_1}^+(x) \cap D_{P_2}^+(x) = D_{P_1 \cup P_2}^+(x)$ and $D_{P_1}^-(x) \cap D_{P_2}^-(x) = D_{P_1 \cup P_2}^-(x)$.

Definition 2.4 ([32]). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, for any $P \subseteq C$, the lower and upper approximations of $Cl_n^{\succeq}$ are respectively defined as follows

$$\underline{P}\left(Cl_n^{\succeq}\right) = \left\{x \in U : D_P^+(x) \subseteq Cl_n^{\succeq}\right\}; \tag{4}$$

$$\overline{P}\left(Cl_n^{\succeq}\right) = \left\{x \in U : D_P^-(x) \cap Cl_n^{\succeq} \neq \emptyset\right\}. \tag{5}$$

The lower and upper approximations of $\mathrm{Cl}_n^{\preceq}$ are respectively defined as follows

$$\underline{P}\left(Cl_n^{\preceq}\right) = \left\{x \in U : D_P^-(x) \subseteq Cl_n^{\preceq}\right\}; \tag{6}$$

$$\overline{P}\left(Cl_n^{\preceq}\right) = \left\{x \in U : D_P^+(x) \cap Cl_n^{\preceq} \neq \emptyset\right\}. \tag{7}$$

Example 2. d in Table 1 is ranked like $C \prec B \prec A$. The approximate sets are $\mathrm{Cl}_1^{\succeq} = \{x_1, x_2, x_3\}$, $\mathrm{Cl}_2^{\succeq} = \{x_1, x_2\}$, and $\mathrm{Cl}_3^{\succeq} = \{x_1\}$, $\mathrm{Cl}_1^{\preceq} = \{x_3\}$, $\mathrm{Cl}_2^{\preceq} = \{x_2, x_3\}$, and $\mathrm{Cl}_3^{\preceq} = \{x_1, x_2, x_3\}$. According to Definition 2.4, the approximations of the upward unions are calculated as $\underline{P}\left(Cl_1^{\succeq}\right) = \{x_1, x_2, x_3\}$, $\underline{P}\left(Cl_2^{\succeq}\right) = \{x_1, x_2\}$, $\underline{P}\left(Cl_3^{\succeq}\right) = \{x_1\}$, $\overline{P}\left(Cl_1^{\succeq}\right) = \{x_1, x_2, x_3\}$, $\overline{P}\left(Cl_2^{\succeq}\right) = \{x_1, x_2\}$, and $\overline{P}\left(Cl_3^{\succeq}\right) = \{x_1\}$. The approximations of the downward unions are calculated as $\underline{P}\left(\mathrm{Cl}_1^{\preceq}\right) = \{x_3\}$, $\underline{P}\left(\mathrm{Cl}_2^{\preceq}\right) = \{x_2, x_3\}$, $\underline{P}\left(\mathrm{Cl}_3^{\preceq}\right) = \{x_1, x_2, x_3\}$, $\overline{P}\left(Cl_1^{\preceq}\right) = \{x_3\}$, $\overline{P}\left(\mathrm{Cl}_2^{\preceq}\right) = \{x_2, x_3\}$, and $\overline{P}\left(\mathrm{Cl}_3^{\preceq}\right) = \{x_1, x_2, x_3\}$.

2.2. The basics of entropy of dominance conditions

At this part, we will introduce some fundamental knowledge about dominance entropy and introduce ordered decision system(ODS) attribute reduction method.

Definition 2.5 ([53]). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, for whatever $A \subseteq C$, the dominance information entropy(DIE) of U about A is defined like this

$$DH_A^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{\left|D_A^+(x_i)\right|}{|U|}. \tag{8}$$

Besides, for arbitrary $A, B \subseteq C$, the DIE of U concerning A and B is defined like this

$$DH_{A\cup B}^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{\left|D_A^+(x_i) \cap D_B^+(x_i)\right|}{|U|} = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{\left|D_{A\cup B}^+(x_i)\right|}{|U|}. \tag{9}$$

Definition 2.6 ([53]). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, for whatever $A \subseteq C$, the dominance conditional entropy(DCE) of A to d is defined like this

$$DH_{d|A}^{\succeq}(U) = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{\left|D_d^+(x_i) \cap D_A^+(x_i)\right|}{\left|D_A^+(x_i)\right|} = -\frac{1}{|U|}\sum_{i=1}^{n} \log \frac{\left|D_{\{d\}\cup A}^+(x_i)\right|}{\left|D_A^+(x_i)\right|}. \tag{10}$$

From Definition 2.6, We can get the hierarchical relation reflected by DCE produces consistent objects, which are closely related to the set of information condition attributes and decision attribute provided.

For attribute reduction methods, we can evaluate the importance of attribute features and the relationship between primary and secondary importance through the concept of attribute importance.

Definition 2.7 ([6], *Attribute Importance Based On In-DCE*). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, $\forall A \subseteq C$ and $\forall a \in A$ hold, the attribute importance based on in-DCE of a in A is defined like this

$$\mathrm{sig}_{\mathrm{inner}}^{\succeq U}(a, A, d) = DH_{d|A-\{a\}}^{\succeq}(U) - DH_{d|A}^{\succeq}(U). \tag{11}$$

Through this definition of attribute importance based on in-DCE. We can select the desired conditional attribute from the entire set of conditional attributes. Besides, important core attribute definitions for attribute condition set A as $Core_A = \left\{a \in A \mid \mathrm{sig}_{inner}^{\succeq U}(a, A, d) > 0\right\}$.

Definition 2.8 ([6], *Attribute Importance Based On Out-DCE*). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, $\forall B \subseteq C$ and $\forall a \in (C - B)$, the attribute importance on the basics of out-DCE of a to B is defined like this

$$\mathrm{sig}_{\mathrm{outer}}^{\succeq U}(a, B, d) = DH_{d|B}^{\succeq}(U) - DH_{d|B\cup\{a\}}^{\succeq}(U). \tag{12}$$

This is similar to the conditional attribute internal importance measure, and the external importance measure can select all necessary conditional attributes except the set of selected conditional attributes.

Definition 2.9 (*Attribute Reduction*). Given $S^{\succeq} = (U, C \cup \{d\}, V, f)$ is an ordered data system, $\forall B \subseteq C$, the conditional attribute subset B is a reduct subset of $S^{\succeq}$ as long as it meets follows

(1) $DH_{d|B}^{\succeq}(U) = DH_{d|C}^{\succeq}(U)$;

(2) $\forall a \in B$, $DH_{d|B-\{a\}}^{\succeq}(U) \neq DH_{d|B}^{\succeq}(U)$.

The above condition (1) is used to ensure that the classification ability of the selected conditional attribute subset is comparable to that of the raw attribute set. Condition (2) is to continuously delete redundant conditional attributes from the selected conditional attribute subset, so as to ensure that the selected conditional attribute subset is not redundant, there is no redundant attribute, and each attribute in the set is indispensable. Thus, if the selected subset of attributes meets both of the above conditions, it is called reduction, otherwise it is called relative reduction.

3. Matrix - based dominant relation reduction method

At this part, we first define and demonstrate the OIS dominance matrix. Then it introduces a calculation method MDCE about matrix DCE. Subsequently, an attribute reduction algorithm on the basics of MDCE is also introduced.

3.1. Matrix based fundamentals of dominance conditional entropy

Definition 3.1. Given $S \geq= (U, AT, V, f)$ is an OIS, for any $A \subseteq AT$, D_A is a dominance relation below A, the dominance relation matrix on U concerning A is described like $\mathbb{M}_U^{\geq A} = \left[m_{(i,j)}^A\right]_{n\times n}$, like this

$$m_{(i,j)}^A = \begin{cases} 1, & x_j D_A x_i; \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Property 3.1. *$\mathbb{M}_U^{\geq A} = \left[m_{(i,j)}^A\right]_{n\times n}$ is a dominance relation matrix, it holds these properties as follow.*

(1) $m_{(i,i)}^A = 1$, where $i \in [1, n]$ and $i \in N^+$;

(2) $\sum_{j=1}^n m_{(i,j)}^A = \left|D_A^+(x_i)\right|$ and $\sum_{i=1}^n m_{(i,j)}^A = \left|D_A^-(x_j)\right|$, where $i, j \in [1, n]$ and $i, j \in N^+$.

Definition 3.2 ("$\cap$"). Given $S^{\geq} = (U, AT, V, f)$ is an OIS, for any $A, B \subseteq AT$, two dominance relation matrices on U concerning A and B are denoted like $\mathbb{M}_U^{\geq A} = \left[m_{(i,j)}^A\right]_{n\times n}$ and $\mathbb{M}_U^{\geq B} = \left[m_{(i,j)}^B\right]_{n\times n}$. Thus "$\cap$" operation between $\mathbb{M}_U^{\geq A}$ and $\mathbb{M}_U^{\geq B}$ is defined like this

$$\mathbb{M}_U^{\geq A} \cap \mathbb{M}_U^{\geq B} = \left[m_{(i,j)}^A \times m_{(i,j)}^B\right]_{n\times n}. \quad (14)$$

From formula (14), we are prone to find how to get a new dominance relation matrix $\mathbb{M}_U^{\geq A}$ and $\mathbb{M}_U^{\geq B}$. Its practical significance lies in the fact that the dominance relation matrix of attribute set can be obtained A and B at the same time.

Proposition 3.1. *Given $S^{\geq} = (U, AT, V, f)$ is an OIS, for any $A, B \subseteq AT$, then $\mathbb{M}_U^{\geq A\cup B} = \mathbb{M}_U^{\geq A} \cap \mathbb{M}_U^{\geq B}$ establishes.*

Proof. From Definition 3.1, $\mathbb{M}_U^{\geq A\cup B} = \left[m_{(i,j)}^{A\cup B}\right]_{n\times n}$. If $m_{(i,j)}^{A\cup B} = 1$, $x_j \in D_{A\cup B}^+(x_i)$. Then, we have $x_j \in D_A^+(x_i)$ and $x_j \in D_B^+(x_i)$, $m_{(i,j)}^A = 1$ and $m_{(i,j)}^B = 1$. Then $m_{(i,j)}^{A\cup B} = m_{(i,j)}^A \times m_{(i,j)}^B = 1$, and vice versa. If $m_{(i,j)}^{A\cup B} = 0$, i.e., $x_j \notin D_{A\cup B}^+(x_i)$, that is, $x_j \notin D_A^+(x_i)$ or $x_j \notin D_B^+(x_i)$, i.e., $m_{(i,j)}^A = 0$ or $m_{(i,j)}^B = 0$. Thus, as we look like $m_{(i,j)}^{A\cup B} = m_{(i,j)}^A \times m_{(i,j)}^B = 0$, it is the same if it is the other way around. Generally speaking, We easily find that from the above that $m_{(i,j)}^{A\cup B} = m_{(i,j)}^A \times m_{(i,j)}^B$, i.e., $\mathbb{M}_U^{\geq A\cup B} = \mathbb{M}_U^{\geq A} \cap \mathbb{M}_U^{\geq B}$ holds.

Definition 3.3. Given $S^{\geq} = (U, AT, V, f)$ is an OIS, for any $A \subseteq AT$, the dominant diagonal relationship matrix $\mathbb{M}_U^{\geq A} = \left[m_{(i,j)}^A\right]_{n\times n}$ is described like $\mathbb{D}_U^{\geq A} = \left[d_{(i,j)}^A\right]_{n\times n}$, like

$$d_{(i,j)}^A = \begin{cases} \sum_{l=1}^n m_{(i,l)}^A, & i, j \in [1, n], i = j; \\ 0, & i, j \in [1, n], i \neq j. \end{cases} \quad (15)$$

Besides, the dominant diagonal relationship matrix of determinant is expressed as $\left|\mathbb{D}_U^{\geq A}\right| = \Pi_{i=j=1}^n d_{ij}^A$, the inverse matrix of the dominant diagonal relationship matrix is represented like $\left(\mathbb{D}_U^{\geq A}\right)^{-1} = \left[\frac{1}{d_{(i,j)}^A}\right]_{n\times n}$, where

$$\frac{1}{d_{(i,j)}^A} = \begin{cases} \frac{1}{\sum_{l=1}^n m_{(i,l)}^A}, & i, j \in [1, n], i = j; \\ 0, & i, j \in [1, n], i \neq j. \end{cases} \quad (16)$$

Corollary 3.1 (*Matrix Dominance Conditional Entropy*). *Given $S^{\geq} = (U, C \cup \{d\}, V, f)$ is an ODS, for any $A \subseteq C$, on the basics of the dominant diagonal relationship matrices $\mathbb{D}_U^{\geq A}$ and $\mathbb{D}_U^{\geq A\cup\{d\}}$, matrix dominance conditional entropy of A to d is like this*

$$MDH_{d|A}^{\geq}(U) = -\frac{1}{|U|} \log \left|\mathbb{D}_U^{\geq A\cup\{d\}} * \left(\mathbb{D}_U^{\geq A}\right)^{-1}\right|. \quad (17)$$

Table 2
An example of ordered decision system.

U	a_1	a_2	a_3	a_4	d
x_1	M	H	F	E	D
x_2	H	L	F	G	B
x_3	L	M	G	E	B
x_4	M	H	P	E	C
x_5	H	L	F	G	A
x_6	L	M	G	E	B
x_7	H	L	F	G	B

Proof. From Definition 2.6, we easily find that $DH_{d|A}^{\succ}(U) = -\frac{1}{|U|} \sum_{i=1}^n \log \frac{\left|D_{\{d\}\cup A}^+(x_i)\right|}{\left|D_A^+(x_i)\right|} = -\frac{1}{|U|} \log \frac{\Pi_{i=1}^n \left|D_{dd|\cup A}^+(x_i)\right|}{\Pi_{i=1}^n \left|D_A^+(x_i)\right|}$. According to Definitions 3.1 and 3.3, the dominance diagonal matrices $\mathbb{D}_U^{\geq A} = \left[d_{(i,j)}^A\right]_{n\times n}$ and $\mathbb{D}_U^{\geq A\cup\{d\}} = \left[d_{(i,j)}^{A\cup\{d\}}\right]_{n\times n}$, where $d_{(i,j)}^A = \left|D_A^+(x_i)\right|$ and $d_{(i,j)}^{A\cup\{d\}} = \left|D_{A\cup\{d\}}^+(x_i)\right|$. Because $\left|\mathbb{D}_U^{\geq A\cup\{d\}} \cdot \left(\mathbb{D}_U^{\geq A}\right)^{-1}\right| = \Pi_{i=1}^n \frac{d_{(i,j)}^{A\cup\{d\}}}{d_{(i,j)}^A} = \frac{\Pi_{i=1}^n d_{(i,j)}^{A\cup|\{d\}}}{\Pi_{i=1}^n d_{(i,j)}^A} = \frac{\Pi_{i=1}^n \left|D_{\{d\}\cup A}^+(x_i)\right|}{\Pi_{i=1}^n \left|D_A^+(x_i)\right|}$. Thus, we can get $DH_{d|A}^{\geq}(U) = MDH_{d|A}^{\geq}(U)$. In short, the results obtained by calculating the dominant conditional entropy by matrix and non-matrix methods are the same.

From formula (17), we find that the core part of MDCE is $\left|\mathbb{D}_U^{\geq A\cup\{d\}} \cdot \left(\mathbb{D}_U^{\geq A}\right)^{-1}\right|$, where the dimensions of the diagonal matrix are clearly and directly shown $\mathbb{D}_U^{\geq A\cup\{d\}}$ to $\mathbb{D}_U^{\geq A}$. Its meaning likes the formula (10). Finally, an example is given to illustrate the calculation method of matrix dominance conditional entropy.

Example 3. Table 2 is a table concerning car evaluation, which meets the various conditions of ODS. As for Table 2, the four conditional attributes are: load-bearing capacity, maximum peak speed, driving experience, and test driver's evaluation of the vehicle. In Table 2, $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$ on behalf of seven cars, $C = \{a_1, a_2, a_3, a_4\}$, where a_1 on behalf of load-bearing capacity, a_2 on behalf of maximum peak speed, a_3 on behalf of driving experience, and a_4 on behalf of test driver's evaluation of the vehicle. The different feature rankings are like this V_{a_1} : L $\prec$ M $\prec$ H, V_{a_2} : L $\prec$ M $\prec$ H, V_{a_3} : P $\prec$ F $\prec$ G, V_{a_4} : F $\prec$ G $\prec$ E, and $V_d : D \prec C \prec B \prec A$.

Through Definition 3.1, the dominance relation matrices $\mathbb{M}_U^{\geq C}$ and $\mathbb{M}_U^{\geq d}$ are as follows

$$\mathbb{M}_U^{\geq C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}_{7\times 7}$$

$$\mathbb{M}_U^{\geq d} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}_{7\times 7}$$

Taking $\mathbb{M}_U^{\geq C}$ as an example, Property 3.1 is verified as follows

(1) For any $i \in [1, 7]$ and $i \in N^+$, $m_{(i,i)}^C = 1$;

(2) For any $i, j \in [1, 7]$ and $i, j \in N^+$, $\sum_{j=1}^7 m_{(i,j)}^C = \left|D_C^+(x_i)\right|$ and $\sum_{i=1}^7 m_{(i,j)}^C = \left|D_C^-(x_j)\right|$, ..., while as $i = 1$, $D_C^+(x_1) = \{x_1\}$,

there is $\sum_{j=1}^{7} m^{C}_{(1,j)} = \left|D^{+}_{C}(x_1)\right| = 1$, while as $j = 1, D^{-}_{C}(x_1) = \{x_1, x_4\}$, we have $\sum_{i=1}^{7} m^{C}_{(i,1)} = \left|D^{-}_{C}(x_1)\right| = 2$.

From Definition 3.2, the dominance relation matrix $\mathbb{M}^{\succeq C\cup\{d\}}_{U}$ is calculated as

$$\mathbb{M}^{\succeq C\cup\{d\}}_{U} = \mathbb{M}^{\succeq C}_{U} \cap \mathbb{M}^{\succeq d}_{U}$$

$$= \begin{bmatrix} 1\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 & 0\times 1 \\ 0\times 0 & 1\times 1 & 0\times 1 & 0\times 0 & 1\times 1 & 0\times 1 & 1\times 1 \\ 0\times 0 & 0\times 1 & 1\times 1 & 0\times 0 & 0\times 1 & 1\times 1 & 0\times 1 \\ 1\times 0 & 0\times 1 & 0\times 1 & 1\times 1 & 0\times 1 & 0\times 1 & 0\times 1 \\ 0\times 0 & 1\times 0 & 0\times 0 & 0\times 0 & 1\times 1 & 0\times 0 & 1\times 0 \\ 0\times 0 & 0\times 1 & 1\times 1 & 0\times 0 & 0\times 1 & 1\times 1 & 0\times 1 \\ 0\times 0 & 1\times 1 & 0\times 1 & 0\times 0 & 1\times 1 & 0\times 1 & 1\times 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}_{7\times 7}.$$

Subsequently, from Definition 3.3, the dominance relation diagonal matrices $\mathbb{D}^{\succeq C}_{U}$, $\mathbb{D}^{\succeq C\cup\{d\}}_{U}$, its inverse matrix $\left(\mathbb{D}^{\succeq C}_{U}\right)^{-1}$ is calculated as

$$\mathbb{D}^{\succeq C}_{U} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix}_{7\times 7},$$

$$\mathbb{D}^{\succeq C\cup\{d\}}_{U} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix}_{7\times 7},$$

$$\left(\mathbb{D}^{\succeq C}_{U}\right)^{-1} = \begin{bmatrix} 1/1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/3 \end{bmatrix}_{7\times 7}.$$

Finally, according to Corollary 3.1, MDCE of C to d could easily get via matrices $\mathbb{D}^{\succeq C\cup\{d\}}_{U}$ and $\left(\mathbb{D}^{\succeq C}_{U}\right)^{-1}$ as $MDH^{\succeq}_{d|C}(U) = -\frac{1}{7}\log\left|\mathbb{D}^{\succeq C\cup\{d\}}_{U}\cdot\left(\mathbb{D}^{\succeq C}_{U}\right)^{-1}\right| = 0.3693$.

Corollary 3.2 (*MDCE-BISM*)**.** *Given $S \succeq = (U, C\cup\{d\}, V, f)$ is an ODS, for any $B \subseteq C$ and $\forall a \in B$, MDCE-BISM of a in B is described like*

$$\mathrm{Msig}^{\succeq U}_{inner}(a, B, d) = MDH^{\succeq}_{d|(B-\{a\})}(U) - MDH^{\succeq}_{d|B}(U). \tag{18}$$

Internal significance measurement on dominance conditional entropy and matrix dominance conditional entropy is consistent, so the same result will be obtained by calculating formula (11) and (18).

Corollary 3.3 (*MDCE-BOSM*)**.** *Given $S^{\succeq} = (U, C\cup\{d\}, V, f)$ is an ODS, for any $B \subseteq C$ and $\forall a \in (C - B)$, MDCE-BOSM of a to B is described like*

$$\mathrm{Msig}^{\succeq U}_{outer}(a, B, d) = MDH^{\succeq}_{d|B}(U) - MDH^{\succeq}_{d|B\cup\{a\}}(U). \tag{19}$$

External significance measures based on dominance conditional entropy and matrix dominance conditional entropy also have the same meaning and are consistent in their calculations by Eqs. (12) and (19).

Algorithm 1: HAR algorithm

Input: An ODS $S^{\succeq} = (U, C\cup\{d\}, V, f)$.
Output: A reduct Red_U.

1 Initialize $\mathrm{Red}_U \leftarrow \emptyset$;
2 Calculate MDCE $MDH^{\succeq C}_{d|C}(U)$ in U via using (17);
3 **for** *h=0 to |C-1|* **do**
4 Calculate $Msig^{\succeq U}_{\text{inner}}(a_k, C, d)$ via using (18);
5 **if** $Msig^{\succeq U}_{inner}(a_k, C, d) > 0$, **then**
6 $\mathrm{Red}_U \leftarrow \mathrm{Red}_U \cup \{a_k\}$;
7 **end**
8 **end**
9 Let B $\leftarrow \mathrm{Red}_U$;
10 **while** $\mathrm{MDH}_{d\mathrm{R}}(U) \neq MDH^{\succeq}_{dr}(U)$ **do**
11 **for** *t=0 to |C-B-1|* **do**
12 Calculate $Msig^{\succeq U}_{\text{outer}}(a_l, B, d)$ via using (19);
13 **end**
14 Select $a_0 = \max\left\{Msig^{\succeq U}_{\text{outer}}(a_l, B, d), a_l \in (C-B)\right\}$;
15 B $\leftarrow B \cup \{a_0\}$
16 **end**
17 **for** *each $a \in B$* **do**
18 **if** $MDH^{\succeq}_{d|[B-\{a\}]}(U) = MDH^{\succeq}_{d|B}(U)$, **then**
19 B $\leftarrow B - \{a\}$;
20 **end**
21 **end**
22 $\mathrm{Red}_U \leftarrow$ B ;
23 **return** Red_U;

3.2. An attribute reduction algorithm HAR related to MDCE

This section will introduce the attribute reduction algorithm associated with MDCE in an ordered data system. As to this algorithm, which will calculate the reduction from scratch when the reduction data object changes and retrain the dynamic ODS to a new reduction. Therefore, compared with the feature incremental algorithm, this algorithm is not a dynamic attribute reduction algorithm, but it lays a foundation for the feature incremental attribute reduction algorithm in the following paper. Here are the steps of Algorithm 1.

The steps in Algorithm 1 are explained in detail as follows. Step 2 Calculate the MDCE of the original ordered information system. The main purpose of steps $3-8$ is to obtain important core attributes and preliminarily get reduced subsets. Steps $10-16$ is mainly to find out whether there are important core attributes from the attributes that have been preliminarily screened out until it is determined that the remaining attributes are all redundant attributes. Steps $17-21$ delete redundant attributes from the existing attribute set to ensure that each attribute in the attribute set is indispensable. Generally speaking, the time complexity of Algorithm 1 is $O\left(|C||U|^2 + |C|^2|U|^2 + |C|^2|U|^2 + |B|^2|U|^2\right)$. Besides, the space complexity is $O\left(|U|^2 + |C||U|^2\right)$.

4. Incremental attribute reduction mechanism under multi-feature change

In an OIS, the characteristics of a data set can be divided into adding features and deleting features. It will take a lot of time and space to calculate the reduction again from scratch after the feature changes. Therefore, in this section, we propose and describe in detail two kinds of feature incremental attribute reduction algorithms, which can make use of the previously obtained reduction results, save a lot of time and space, and greatly reduce the time and space complexity of the algorithm.

4.1. An incremental multi-objective feature attribute reduction method when adding features

At this part, we first introduce how MDCE is updated when multiple features are added. Then, we introduce and illustrate the updating algorithm of attribute reduction in ordered information system.

4.1.1. MDCE update principle when adding attribute features

This section describes the incremental update method based on the dominance relationship matrix for computing a new MDCE when many features are added to the ordered information system. The key of this updating principle is how to use the predominance relation matrix and the predominance diagonal matrix, which are introduced as follows.

Proposition 4.1 (*Dominance Relation Matrix*)**.** *There is an OIS $S^{\succeq} = (U, AT, V, f)$, where $A = \{x_1, x_2, \ldots, x_n\}$.$\forall U \subseteq U$, assume as the dominance relation matrix on U concerning A is $\mathbb{M}_U^{\succeq A} = \left[m_{(i,j)}^A\right]_{n\times n}$, the feature set $A^+ = \{x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$ is added to $S^{\succeq}$. The updated dominance relation matrix on U concerning $A \cup A^+$ is like the $\mathbb{M}_U^{\succeq A\cup A^+} = \left[m_{(i,j)}^{\prime A\cup A^+}\right]_{n\times n}$, as follows*

$$m_{(i,j)}^{\prime A\cup A^+} = \begin{cases} 1, & U_j(A\cup A^+) \geqq U_i(A\cup A^+); \\ 0, & otherwise. \end{cases} \tag{20}$$

Proposition 4.1 offers this rationale to update the dominance relation matrix while as multiple features are added. The basic idea is to judge whether the newly added conditional attributes of the original dominant object are still dominant on the basis of a new dominant relation matrix is obtained by updating the original matrix. Examples are as follows.

Example 4. A new feature attribute set is added based on Table 1. $C^+ = \{a5\}$, $a5 = \{l, m, m, l, h, m, m\}$, after this process, the original dominance relation matrix $M_U^{\succeq C}$ and the new conditional attribute dominance relation matrix $M_U^{\succeq C\cup C^+}$ can be expressed as

$$\mathbb{M}_U^{\succeq C} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}$$

$$\begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&\mathbf{1}&0&0&1&0&\mathbf{1}\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix} \rightarrow \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&\mathbf{0}&0&0&1&0&\mathbf{0}\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}$$

$$\mathbb{M}_U^{\succeq C\cup C^+} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}.$$

Proposition 4.2 (*Dominant Realtionship Diagonal Matrix*)**.** *There is an OIS $S^{\succeq} = (U, AT, V, f)$, where $A = \{x_1, x_2, \ldots, x_n\}$, For any $A \subseteq AT$, then the dominant realtionship diagonal matrix on U concerning A is $\mathbb{D}_U^{\succeq A} = \left[d_{(i,j)}^A\right]_{n\times n}$, the feature set $A^+ = \{x_{n+1}, x_{n+2}, \ldots, x_{n+n'}\}$ is added to $S^{\succeq}$. The updated dominant realtionship diagonal matrix on U concerning $A \cup A^+$ is like the $\mathbb{D}_U^{\succeq A\cup A^+} = \left[d_{(i,j)}^{\prime A\cup A^+}\right]_{n\times n}$, where*

$$d_{(i,j)}^{\prime A\cup A^+} = \begin{cases} d_{(i,j)}^A - m_{(i,j)}^{\prime A\cup A^+}, & U_j(A\cup A^+) \geq U_i(A\cup A^+); \\ d_{(i,j)}^A, & U_j(A\cup A^+) < U_i(A\cup A^+). \end{cases} \tag{21}$$

Example 5. Continuing from Example 4, known matrices $\mathbb{M}_U^{\succeq C\cup C^+}$ and $\mathbb{D}_U^{\succeq C}$, we can update matrix $\mathbb{D}_U^{\succeq C\cup C^+}$ by using Proposition 4.2 as

$$\mathbb{D}_U^{\succeq C\cup C^+} = \begin{bmatrix} 1-0&0&0&0&0&0&0\\ 0&3-0&0&0&0&0&0\\ 0&0&2-0&0&0&0&0\\ 0&0&0&2-0&0&0&0\\ 0&0&0&0&3-2&0&0\\ 0&0&0&0&0&2-0&0\\ 0&0&0&0&0&0&3-0 \end{bmatrix} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&3&0&0&0&0&0\\ 0&0&2&0&0&0&0\\ 0&0&0&2&0&0&0\\ 0&0&0&0&1&0&0\\ 0&0&0&0&0&2&0\\ 0&0&0&0&0&0&3 \end{bmatrix}_{7\times 7}.$$

Next, we will walk through the detailed steps of how to calculate a new MDCE after adding multiple attribute characteristics. As for any $X \subseteq U$, we have known the raw matrices are $\mathbb{M}_U^{\succeq A}$, $\mathbb{M}_{U\cup X^+}^{\succeq A}$, $\mathbb{D}_U^{\succeq A}$, and $\mathbb{D}_{U\cup X^+}^{\succeq A}$. When A^+ is added to $S^{\succeq}$, through the Propositions 4.1 and 4.2, we could easily gain the updated dominant realtionship diagonal matrices $\mathbb{D}_U^{\succeq A\cup A^+}$ and $\mathbb{D}_{U\cup X^+}^{\succeq A}$. So, we can effortlessly calculate the MDCE $MDH_{d|U}^{\succeq}\left(A\cup A^+\right)$ by Corollary 3.1.

4.1.2. A dynamic incremental attribute feature reduction algorithm when adding features

In Algorithm 2, inspired by the updating principle of MDCE, a multi-feature incremental attribute reduction algorithm (HAR-A) is presented. The detailed steps of Algorithm 2 are as follows.

The steps of Algorithm 2 are described in detail. Steps 2–4 incrementally calculate the new dominance relation matrix and its dominance relation diagonal matrix using this method in Propositions 4.1 and 4.2. Step 5 calculate the updated MDCE via the Corollary 3.1. Steps 6–10 is mainly to determine whether the new MDCE is equal to the MDCE of the original attribute subset (that is, the raw reduction) as same as the new MDCE below this whole attribute set. If in case, leave the raw subset of attributes will not change. Steps 11–16 Arrange the eliminated attributes in descending order to form a new set, and update the selected attribute subset until the end of Step 12.

Algorithm 2: HAR-A algorithm

Input:
(1) A raw ODS $S^{\succeq} = (U, C \cup \{d\}, V, f)$, $whereC = \{a_1, a_2, \ldots, a_n\}$, $C^+ = \{a_{n+1}, a_{n+2}, \ldots, a_{n+n}\}$;
(2) The original reduct Red_U on U;
(3) The original dominance relation matrices $M_U^{\succeq C} = \left[m_{(i,j)}^{C}\right]_{n\times n}$, $M_U^{\succeq \mathrm{Red}_U} = \left[m_{(i,j)}^{Red_U}\right]_{n\times n}$, and$M_U^{\succeq d} = \left[m_{(i,j)}^{d}\right]_{n\times n}$;
(4) The original dominance diagonal matrices $D_U^{\succeq C} = \left[d_{(i,j)}^{C}\right]_{n\times n}$, $D_U^{\succeq C\cup\{d\}} = \left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $D_U^{\succeq \mathrm{Red}_U} = \left[d_{(i,j)}^{\mathrm{Red}_U}\right]_{n\times n}$ and$D_U^{\succeq \mathrm{Red}_U\cup\{d\}} = \left[d_{(i,j)}^{\mathrm{Red}_U\cup\{d\}}\right]_{n\times n}$.
Output: A new reduct $\mathrm{Red}_{U'}$.

1 Initialize $B \leftarrow \mathrm{Red}_U$, $C' \leftarrow C \cup C^+$, $M_U^{\succeq C'} \leftarrow M_U^{\succeq C}$, $D_U^{\succeq C'} \leftarrow D_U^{\succeq C}$, $D_U^{\succeq C'\cup\{d\}} \leftarrow D_U^{\succeq C\cup\{d\}}$;
2 Compute new dominance relation matrices $M_U^{\succeq C'} \leftarrow \left[m_{(i,j)}^{\prime C}\right]_{n\times n}$, $M_U^{\succeq B} \leftarrow \left[m_{(i,j)}^{\prime B}\right]_{n\times n}$, $M_U^{\succeq d} \leftarrow \left[m_{(i,j)}^{\prime d}\right]_{n\times n}$ via using Proposition 4.1;
3 Compute dominance relation matrices $M_U^{\succeq C'\cup\{d\}}$ and $M_U^{\succeq B\cup\{d\}}$;
4 Compute new dominance diagonal matrices $D_U^{\succeq C} = \left[d_{(i,j)}^{C}\right]_{n\times n}$, $D_U^{\succeq C\cup\{d\}} = \left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $D_U^{\succeq \mathrm{Red}_U} = \left[d_{(i,j)}^{\mathrm{Red}_U}\right]_{n\times n}$ and $D_U^{\succeq B\cup\{d\}} \leftarrow \left[d_{(i,j)}^{\prime B\cup\{d\}}\right]_{n\times n}$ via using Proposition 4.2;
5 Compute new MDCE MDCE $MDH_{dC'}^{\succeq}(U)$ and $MDH_{d|B}^{\succeq}(U)$;
6 **if** $MDH_{dC'}^{\succeq}(U) = MDH_{d|B}^{\succeq}(U)$, **then**
7 go to step17;
8 **else**
9 go to step11;
10 **end**
11 For each $a \in (C' - B)$, compute $Msig_{\text{outer}}^{\succeq U}(a, B, d)$, then save the result as $\left\{a'_0, a'_1, \ldots, a_{|c'-B|}\right\}$;
12 **while** $MDH_{d|C'}^{\succeq}(U) \neq MDH_{d|B}^{\succeq}(U)$ **do**
13 **for** $z = 0$ *to* $|C - B - 1|$ **do**
14 Select $B \leftarrow B \cup \left\{a'_z\right\}$, then calculate $MDH_{d|B}^{\succeq}(U)$;
15 **end**
16 **end**
17 **for** *each* $a \in B$ **do**
18 calculate $MDH_{d(B-\{a\})}^{\succeq}(U)$;
19 **if** $MDH_{d|[B-\{a\})}^{\succeq}(U) = MDH_{d|B}^{\succeq}(U)$, **then**
20 $B \leftarrow B - \{a\}$;
21 **end**
22 **end**
23 $\mathrm{Red}_{U'} \leftarrow B$;
24 **return** $\mathrm{Red}_{U'}$;

Steps 17–22 delete redundant attributes from the existing attribute set to ensure that each attribute in the attribute set is indispensable. Steps 23–24 The final reduction result is displayed. Generally speaking, the spatial complexity of Algorithm 2 is $O\left(|U|^2 + \left(\left|C'\right| - |B|\right) U\right|^2\right)$. The time complexity of Algorithm 2 is $O\left(UU \left|C^+\right| CC' \left|+ \left(\left|C'\right| - |B|\right) U\right|^2 + |B|^2|U|^2\right)$. We also compare the complexity of HAR algorithm and HAR-A algorithm, and the results are shown in Table 3.

As can be seen from Table 3, both the time and space complexity of HAR-A algorithm is smaller than that of HAR algorithm. This is because HAR algorithm recalculates the reduction from the beginning when the features change, while HAR-A algorithm inherits the previous reduction results, thus greatly reducing the time and space complexity of the algorithm. Therefore, HAR-A algorithm can save much time in the reduction calculation of large-scale data.

4.2. An incremental multi-objective feature attribute reduction method when deleting features

At this part, we first introduce how MDCE is updated when multiple features are deleted. Then, we introduce and illustrate the updating algorithm of attribute reduction in ordered information system.

4.2.1. MDCE update principle when deleting attribute features

This section describes the incremental update method based on the dominance relationship matrix for computing a new MDCE when many features are deleted from the ordered information system. The key of this updating principle is how to use the predominance relation matrix and the predominance diagonal matrix, which are introduced as follows.

Proposition 4.3 (*Dominance Relation Matrix*)**.** *There is an OIS $S^{\succeq} = (U, AT, V, f)$, where $A = \{x_1, x_2, \ldots, x_n\}$.$\forall U \subseteq U$, suppose that the dominance relation matrix on U concerning A is $\mathbb{M}_U^{\succeq A} = \left[m_{(i,j)}^{A}\right]_{n\times n}$, the feature set $A^- = \left\{x_{q1}, x_{q2}, \ldots, x_{qn'}\right\}$ is deleted from $S^{\succeq}$. The updated dominance relation matrix on U concerning $A \cup A^-$ is like the $\mathbb{M}_U^{\succeq A-A^-} = \left[m_{(i,j)}^{\prime A-A^-}\right]_{n\times n}$, where*

$$m_{(i,j)}^{\prime A-A^-} = \begin{cases} 1, & U_j(A - A^-) \geqq U_i(A - A^-); \\ 0, & otherwise. \end{cases} \tag{22}$$

Example 6. A new feature attribute set is deleted from Table 1. $C^- = \{a3, a4\}$, $a3 = \{f, f, g, p, f, g, f\}$, $a4 = \{e, g, e, e, g, e, g\}$ and the raw dominance relation matrix $M_U^{\succeq C}$ and the new conditional attribute dominance relation matrix $M_U^{\succeq C-C^-}$ can be expressed as

$$\mathbb{M}_U^{\succeq C} = \begin{bmatrix} 1&0&0&0&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 1&0&0&1&0&0&0\\ 0&1&0&0&1&0&1\\ 0&0&1&0&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}$$

$$\begin{bmatrix} 1&0&0&\mathbf{0}&0&0&0\\ 0&1&0&0&1&0&1\\ \mathbf{0}&0&1&\mathbf{0}&0&1&0\\ 1&0&0&1&0&0&0\\ 0&1&0&0&1&0&1\\ \mathbf{0}&0&1&\mathbf{0}&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix} \rightarrow \begin{bmatrix} 1&0&0&\mathbf{1}&0&0&0\\ 0&1&0&0&1&0&1\\ \mathbf{1}&0&1&\mathbf{1}&0&1&0\\ 1&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ \mathbf{1}&0&1&\mathbf{1}&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}$$

$$\mathbb{M}_U^{\succeq C-C^-} = \begin{bmatrix} 1&0&0&1&0&0&0\\ 0&1&0&0&1&0&1\\ 1&0&1&1&0&1&0\\ 1&0&0&1&0&0&0\\ 0&0&0&0&1&0&0\\ 1&0&1&1&0&1&0\\ 0&1&0&0&1&0&1 \end{bmatrix}_{7\times 7}.$$

Proposition 4.4 (*Dominant Realtionship Diagonal Matrix*)**.** *There is an OIS $S^{\succeq} = (U, AT, V, f)$, where $A = \{x_1, x_2, \ldots, x_n\}$, For any A*

Table 3
Complexity comparison of HAR algorithm and HAR-A algorithm.

Algorithm	*HAR*	*HAR − A*
Time complexity	$O\left(\lvert C'\rvert\lvert U\rvert^2+\lvert C'\rvert^2\lvert U\rvert^2+\lvert C'\rvert^2\lvert U\rvert^2+\lvert B\rvert^2\lvert U\rvert^2\right)$	$O\left(\lvert U\rvert\left\lvert C^+\right\rvert\left\lvert C'\right\rvert+(\lvert C'\rvert-\lvert B\rvert)\lvert U\rvert^2+\lvert B\rvert^2\lvert U\rvert^2\right)$
Space complexity	$O\left(\lvert U\rvert^2+\lvert C'\rvert\lvert U\rvert^2\right)$	$O\left(\lvert U\rvert^2+(\lvert C'\rvert-\lvert B\rvert)\lvert U\rvert^2\right)$

$\subseteq AT$, let the dominant realtionship diagonal matrix on U concerning A is $\mathbb{D}_U^{\succeq A}=\left[d_{(i,j)}^A\right]_{n\times n}$, then feature set $A^-=\left\{x_{q1},x_{q2},\ldots,x_{qn'}\right\}$ is deleted from $S^{\succeq}$. The updated dominant realtionship diagonal matrix on U concerning $A-A^-$ is like the $\mathbb{D}_U^{\succeq A-A^-}=\left[d_{(i,j)}^{\prime A-A^-}\right]_{n\times n}$, while as

$$d_{(i,j)}^{\prime A-A^-}=\begin{cases}d_{(i,j)}^A+m_{(i,j)}^{\prime A-A^-}, & U_j(A-A^-)\geqq U_i(A-A^-);\\ d_{(i,j)}^A, & U_j(A-A^-)<U_i(A-A^-).\end{cases}\tag{23}$$

Example 7. Continuing from Example 6, known matrices $\mathbb{M}_U^{\succeq C-C^-}$ and $\mathbb{D}_U^{\succeq C}$, we can update matrix $\mathbb{D}_U^{\succeq C-C^-}$ by using Proposition 4.4 as

$$\mathbb{D}_U^{\succeq C-C^-}$$

$$=\begin{bmatrix}1+1&0&0&0&0&0&0\\0&3+0&0&0&0&0&0\\0&0&2+2&0&0&0&0\\0&0&0&2+0&0&0&0\\0&0&0&0&3+0&0&0\\0&0&0&0&0&2+2&0\\0&0&0&0&0&0&3+0\end{bmatrix}$$

$$=\begin{bmatrix}2&0&0&0&0&0&0\\0&3&0&0&0&0&0\\0&0&4&0&0&0&0\\0&0&0&2&0&0&0\\0&0&0&0&3&0&0\\0&0&0&0&0&4&0\\0&0&0&0&0&0&3\end{bmatrix}_{7\times7}.$$

4.2.2. A dynamic incremental attribute feature reduction algorithm when deleting features

In Algorithm 3, inspired by the updating principle of MDCE, a multi-feature incremental attribute reduction algorithm (HAR-D) is presented. The detailed steps of Algorithm 3 are as follows.

The steps of Algorithm 3 are described in detail. Steps 2–3 incrementally calculate the new dominance relation matrix and its dominant realtionship diagonal matrix using this method in Propositions 4.3 and 4.4. Step 4 calculates new MDCE via the Corollary 3.1. Steps 5–9 is mainly to determine whether the new MDCE is equal to the MDCE of the original attribute subset (that is, the raw reduction) as same as the new MDCE below this whole attribute set. If in case, leave the raw subset of attributes will not change. Steps 10–15 Arrange the eliminated attributes in descending order to form a new set, and update the selected attribute subset until the end of Step 11. Steps 16–21 delete redundant attributes from the existing attribute set to ensure that each attribute in the attribute set is indispensable. Steps 22–23 The final reduction result is displayed. Generally speaking, the time complexity of Algorithm 3 is $O\left(\lvert U\rvert+\left(\lvert C'\rvert-\lvert B\rvert\right)\lvert U\rvert^2+\lvert B\rvert^2\lvert U\rvert^2\right)$. Besides, the space complexity of Algorithm 3 is $O\left(\lvert U\rvert^2+\left(\lvert C'\rvert-\lvert B\rvert\right)\lvert U\rvert^2\right)$. We also compare the complexity of HAR algorithm and HAR-D algorithm, and the results are shown in Table 4.

As can be seen from Table 4, both the time and space complexity of HAR-D algorithm is smaller than that of HAR algorithm. This is because HAR algorithm recalculates the reduction from the beginning when the features change, while HAR-D algorithm

Algorithm 3: HAR-D algorithm

Input:
(1) A raw ODS $S^{\succeq}=(U,C\cup\{d\},V,f),\ where C=\{a_1,a_2,\ldots,a_n\},C^-=\left\{a_{q1},a_{q2},\ldots,a_{qn'}\right\}$ is an deleted feature set ;
(2) The original reduct $Red_U onU$;
(3) The original dominance relation matrices $D_U^{\succeq C}=\left[d_{(i,j)}^C\right]_{n\times n}$, $D_U^{\succeq C\cup\{d\}}=\left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $D_U^{\succeq \mathrm{Red}_U}=\left[d_{(i,j)}^{\mathrm{Red}_U}\right]_{n\times n}$ and$M_U^{Red_U\cup\{d\}}=\left[m_{(i,j)}^{Red_U\cup\{d\}}\right]_{n\times n}$;
(4) The original dominance diagonal matrices $D_U^{\succeq C}=\left[d_{(i,j)}^C\right]_{n\times n}$, $D_U^{\succeq C\cup\{d\}}=\left[d_{(i,j)}^{C\cup\{d\}}\right]_{n\times n}$, $D_U^{\succeq \mathrm{Red}_U}=\left[d_{(i,j)}^{\mathrm{Red}_U}\right]_{n\times n}$ and$D_U^{\succeq \mathrm{Red}_U\cup\{d\}}=\left[d_{(i,j)}^{\mathrm{Red}_U\cup\{d\}}\right]_{n\times n}$.
Output: A new reduct $\mathrm{Red}_{U'}$.

1 Initialize
$B\leftarrow \mathrm{Red}_U$, $C'\leftarrow C-C^-$, $M_U^{\succeq C'}\leftarrow M_U^{\succeq C}$, $M_U^{\succeq C'\cup\{d\}}\leftarrow M_U^{\succeq C\cup\{d\}}$, $D_U^{\succeq C'}\leftarrow D_U^{\succeq C}$, $D_U^{\succeq C'\cup\{d\}}\leftarrow D_U^{\succeq C\cup\{d\}}$;
2 Compute new dominance relation matrices
$M_U^{\succeq C'}\leftarrow\left[m_{(i,j)}^{\prime C}\right]_{n\times n}$, $M_U^{\succeq B}\leftarrow\left[m_{(i,j)}^{\prime B}\right]_{n\times n}$, $M_U^{\succeq C'\cup\{d\}}\leftarrow\left[m_{(i,j)}^{\succeq C'\cup\{d\}}\right]_{n\times n}$ and $M_U^{\succeq B'\cup\{d\}}\leftarrow\left[m_{(i,j)}^{\succeq B'\cup\{d\}}\right]_{n\times n}$ via using Proposition 4.3 ;
3 Compute new dominance diagonal matrices
$D_U^{\succeq C'}\leftarrow\left[d_{(i,j)}^{\prime C}\right]_{n\times n}$, $D_U^{\succeq C'\cup\{d\}}\leftarrow\left[d_{(i,j)}^{\prime C\cup\{d\}}\right]_{n\times n}$, $D_U^{\succeq B}\leftarrow\left[d_{(i,j)}^{\prime B}\right]_{n\times n}$, $D_U^{\succeq B\cup\{d\}}\leftarrow\left[d_{(i,j)}^{\prime B\cup\{d\}}\right]_{n\times n}$ via using Proposition 4.4;
4 Calculate new MDCE MDCE $MDH_{dC'}^{\succeq}(U)$ and $MDH_{d|B}^{\succeq}(U)$;
5 **if** $MDH_{dC'}^{\succeq}(U)=MDH_{d|B}^{\succeq}(U)$, **then**
6 | go to step16;
7 **else**
8 | go to step10;
9 **end**
10 For each $a\in\left(C'-B\right)$, calculate $Msig_{\mathrm{outer}}^{\succeq U}(a,B,d)$, then save the result as $\left\{a_0',a_1',\ldots,a_{|c'-B|}\right\}$;
11 **while** $MDH_{d|C'}^{\succeq}(U)\succeq MDH_{d|B}^{\succeq}(U)$ **do**
12 | **for** $z=1$ *to* $|C-B|$ **do**
13 | | Select $B\leftarrow B\cup\left\{a_z'\right\}$ then calculate $MDH_{d|B}^{\succeq}(U)$;
14 | **end**
15 **end**
16 **for** *each* $a\in B$ **do**
17 | calculate $MDH_{d(B-\{a\})}^{\succeq}(U)$;
18 | **if** $MDH_{d|[B-\{a\})}^{\succeq}(U)=MDH_{d|B}^{\succeq}(U)$, **then**
19 | | $B\leftarrow B-\{a\}$;
20 | **end**
21 **end**
22 $\mathrm{Red}_{U'}\leftarrow B$;
23 **return** $\mathrm{Red}_{U'}$;

Table 4
Complexity comparison of HAR algorithm and HAR-D algorithm.

Algorithm	*HAR*	*HAR − D*
Time complexity	$O\left(\|C'\|\,\|U\|^2+\|C'\|^2\,\|U\|^2+\|C'\|^2\,\|U\|^2+\|B\|^2\,\|U\|^2\right)$	$O\left(\|U\|+\ (\|C'\|-\|B\|)\,\|U\|^2+\|B\|^2\,\|U\|^2\right)$
Space complexity	$O\left(\|U\|^2+\|C'\|\,\|U\|^2\right)$	$O\left(\|U\|^2+(\|C'\|-\|B\|)\,\|U\|^2\right)$

Table 5
The description of datasets.

No.	Datasets	Abbreviation	Objects	Attributes	Classes
1	Zoo	Zoo	101	17	7
2	Breast Cancer Coimbra	Bcc	116	9	2
3	Wine	Wine	178	13	3
4	Hill_valley	Hill	606	100	2
5	Abalone	Abalone	4177	8	3
6	Codon_usage	Codon	13 028	68	10
7	Dry Bean	Bean	13 611	16	8
8	EEG Eye State	Eye	14 980	14	2
9	Letter-recognition	Letter	20 000	16	26

inherits the previous reduction results, thus greatly reducing the time and space complexity of the algorithm. Therefore, HAR-D algorithm can save much time in the reduction calculation of large-scale data.

5. Experimental analysis

In this section, we conduct a series of experiments to prove the effectiveness, efficiency and the performance evaluation of the proposed incremental algorithm for attribute features. A summary of the nine data sets from the UCI used in these experiments is shown in Table 5. In this article, all algorithms are coded by Python using an environment of Anaconda Navigator, and run on a computer with a 2.90 GHz CPU AMD Ryzen 7 4800H with Radeon Graphics, 8.0 GB of memory, and a 64-bit Windows 10 operating system.

At this part, we will evaluate the performance of our proposed algorithm, so we conducted a comparison experiment, and compared the proposed HAR-A algorithm and HAR-D algorithm with the existing four attribute reduction algorithms HAR, DRSQR, FEAR and NRSAR. HAR algorithm is an attribute reduction algorithm based on dominant conditional entropy mentioned above. DRSQR is a fast reduction algorithm on the basis of dominant rough set. FEAR algorithm is an attribute reduction algorithm on the basis of fuzzy entropy. NRSAR algorithm is a neighborhood entropy attribute reduction algorithm on the basis of neighborhood rough sets. In addition, we also use four classifiers BayesNet, RandomTree, Knn and Adaboost to test the effect of classification accuracy of the reduction. We also used 10-classification cross validation.

5.1. Performance evaluation of HAR-A algorithm

At this part, we analyze algorithm HAR-A from classification accuracy, algorithm efficiency and index performance evaluation. The specific design is as follows.

5.1.1. Algorithm classification accuracy comparison

At this part, the classification accuracy of the HAR-A algorithm proposed in this paper is compared with the other four algorithms. From every data set in Table 5, 50% features are randomly selected as the raw feature set, and the left 50% will be the added features. Algorithms HAR-A, HAR, DRSQR, FEAR, and NRSAR are used to compute fresh reductions while as the left 50% features are added to the raw 50% feature set. The experimental results are shown in Tables 6 and 7, where "raw" represents the classification accuracy of the raw attribute set. Note that in Table 6, the numbers in parentheses after each classification precision result represent the size of the reduced set under this condition. Tables 7, 10, and 11 have a frame similar to Table 6.

As shown in the above chart, the classification accuracy of algorithm HAR-A is almost higher than that of other algorithms in all cases, and its average score is far ahead, so the classification accuracy of HRA-A algorithm is very high.

5.1.2. Algorithm efficiency comparison

At this part, we test the efficiency of the algorithm HAR-A and compare it with the other four algorithms in terms of calculation time and acceleration ratio. For each data set in Table 5, five test sets were built. First, 50% of the features are randomly selected as the raw feature set. We then randomly add features from the remaining 50% to the raw feature set to get a dynamic data set to test (that is, randomly select 10%, 20%, 30%, 40%, and 50% of the remaining 50% features and add them to the original feature set). In particular, since the number of attribute features of BCC and Abalone data sets is less than 10 (9 and 8 respectively), the raw data sets of BCC and Abalone data sets are selected to be 4 and 3 respectively, and then one attribute feature is added each time. The time spent using different algorithms on these data sets is then compared. Fig. 2 shows the detailed variation trend of these five algorithms when the characteristics of different data sets change. The abscissa stands for the size of the feature set added, and the ordinate stands for the computation time.

We can see from Fig. 2 that the computation time of these five algorithms will increase with the constant increase of attribute feature set. It can be seen from each subgraph that the computation time of algorithm HAR-A is significantly less than that of other algorithms. Especially for large data sets, the algorithm HAR-A has a very obvious time-saving effect. Therefore, we can conclude that the efficiency of algorithm HAR-A is very high.

Then, we prove the validity of the algorithm HAR-A again from the perspective of acceleration ratio. Based on the results shown in Fig. 2, we calculated the acceleration ratio of the algorithm HAR-A compared with the other four algorithms. The experimental results are shown in Fig. 3. The x-coordinate represents the size of the feature set added, and the y-coordinate stands for the value of the acceleration ratio. These algorithms have high speed ratios for different data sets. Again, in that case, the curve might be so dense that it is hard to see what the trend is. In order to solve the problem, we present the result in three dimensions. For example, Fig. 3(a) the X-axis represents the size of the feature set added. The Y-axis represents data sets, Zoo, Bcc, Wine, Hill, the experimental results value range of Abalone fitting together as the result of the experiment shows the data set, value range of [0, 3]. The Z-axis represents the experimental results fitted together by data sets Codon,Bean,Eye, and Letter. The value range is as follows: the subgraph in Fig. 3(a) shows the experimental results of data sets Zoo, Bcc, Wine, and Hill, with the value range of [0,200]. Figs. 3(b), (c), (d), 5(a) and (b), (c), (d) of the structure is similar to Fig. 3(a).

As can be seen from Fig. 3, the acceleration ratio of algorithm HAR-A against other algorithms in all data sets is more than 0. This indicates that algorithm HAR-A is faster than the other four algorithms on all experimental data sets. Besides, for relatively large data sets, algorithm HAR-A is tens or even hundreds of times faster than the other four algorithms. The above proves again that the efficiency of algorithm HAR-A is very high.

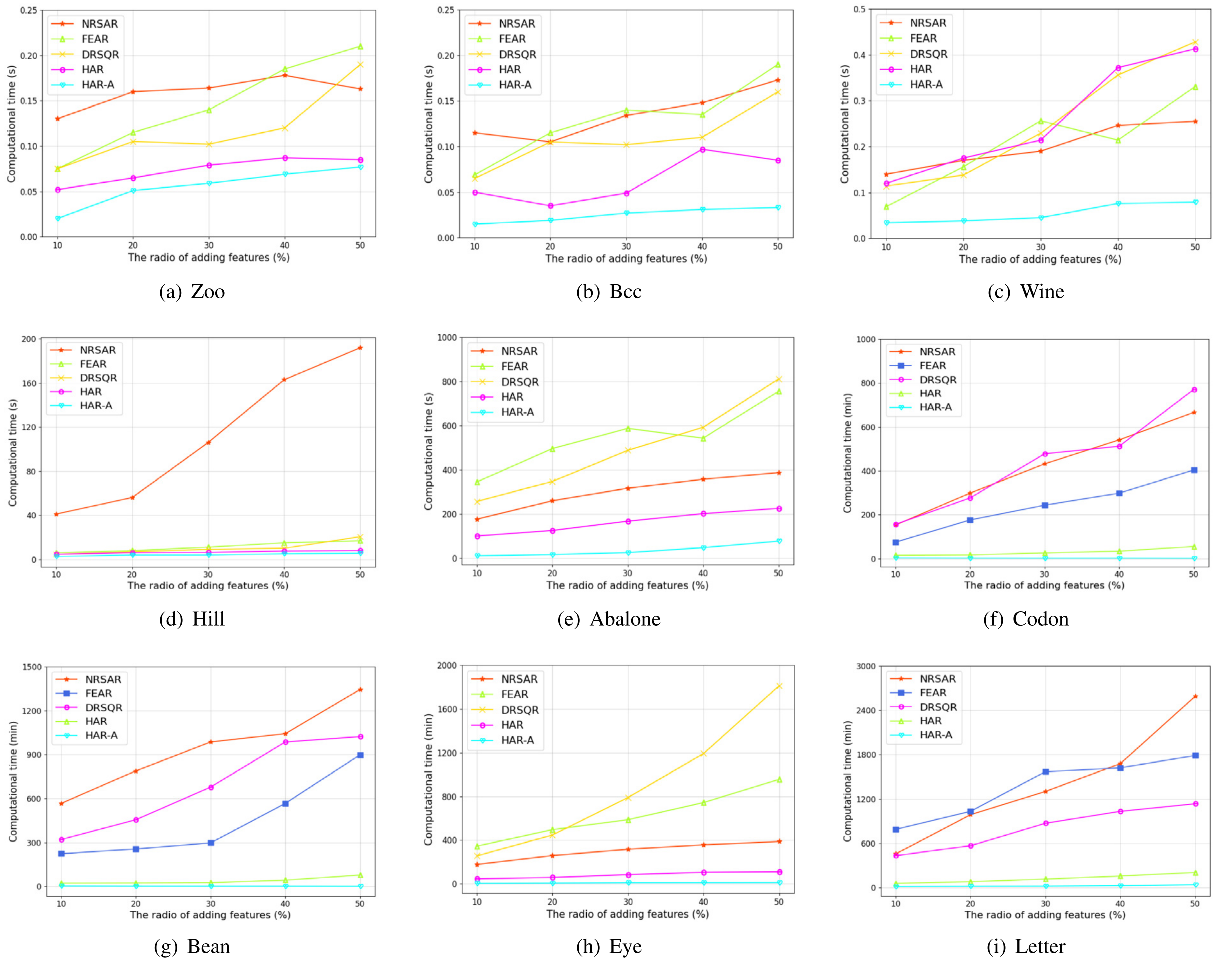


Fig. 2. The computational time of different algorithms versus different ratios of adding features.

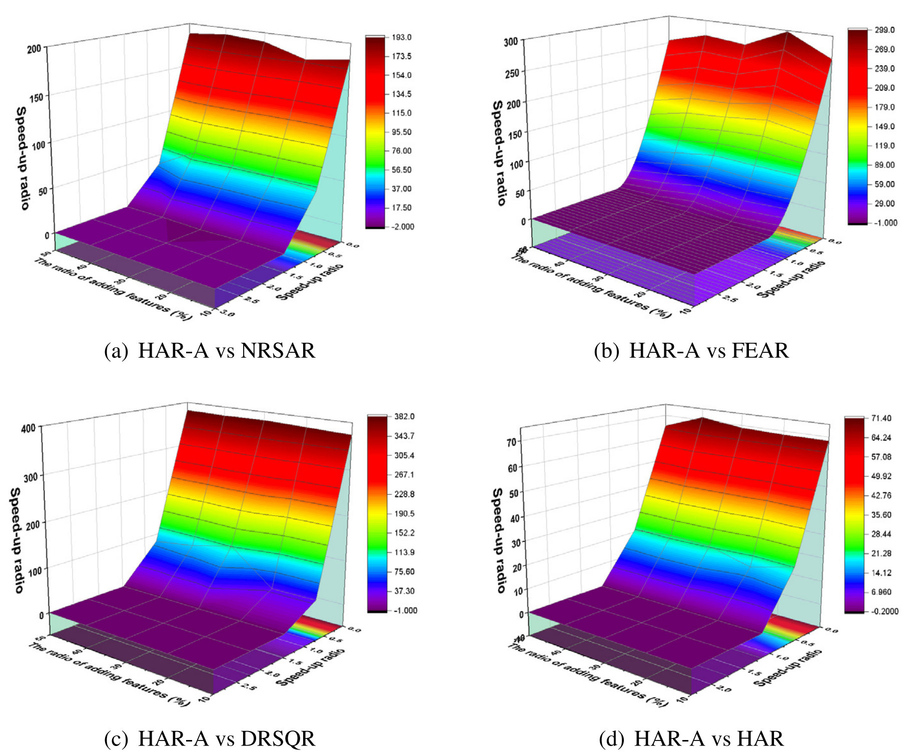


Fig. 3. The speed-up ratios that algorithm HAR-A relates to different algorithms.

Table 6
The comparison of classification accuracies of different algorithms on Bayes Net and Random Tree (%).

Datasets	BayesNet						RandomTree					
	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-A	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-A
Zoo	73.56	74.56(12)	77.23(13)	67.32(7)	76.33(12)	**81.47(15)**	89.91	82.88(12)	87.69(13)	81.47(7)	86.45(12)	**92.13(15)**
Bcc	64.21	61.37(4)	64.98(5)	57.42(9)	63.25(5)	**65.71(6)**	80.03	81.17(4)	69.98(5)	**85.17(9)**	66.74(5)	82.22(6)
Wine	98.15	81.30(2)	85.44(11)	92.34(10)	**98.15(12)**	**98.15(12)**	84.90	86.89(2)	66.81(11)	87.19(10)	79.28(12)	**93.47(12)**
Hill	70.58	54.82(36)	60.21(44)	**72.35(39)**	63.21(44)	62.13(45)	79.43	71.15(36)	80.49(44)	83.37(39)	79.57(44)	**84.27(45)**
Abalone	51.27	57.02(3)	54.11(2)	53.32(3)	51.82(1)	**57.32(2)**	80.07	82.15(3)	74.13(2)	79.81(3)	86.25(1)	**88.76(2)**
Codon	76.34	64.92(11)	72.18(13)	61.40(23)	75.72(14)	**75.81(8)**	77.4	72.89(11)	71.54(13)	80.84(23)	71.08(14)	**84.66(8)**
Bean	70.13	68.45(5)	64.28(4)	73.82(5)	73.13(6)	**78.05(6)**	80.13	79.13(5)	73.20(4)	78.56(5)	79.57(6)	**83.92(6)**
Eye	81.74	77.63(10)	73.19(8)	67.25(8)	88.75(5)	**92.36(6)**	60.11	55.39(10)	54.18(8)	57.62(8)	59.38(5)	**61.25(6)**
Letter	73.28	69.97(8)	71.11(9)	69.77(8)	71.92(5)	**74.71(6)**	79.77	71.89(8)	74.13(9)	77.93(8)	78.19(5)	**82.46(6)**
Average	73.25	67.78	69.19	68.33	73.59	**76.19**	79.10	76.61	72.21	79.11	76.28	**83.68**

Table 7
The comparison of classification accuracies of different algorithms on Knn and Adaboost (%).

Datasets	Knn						Adaboost					
	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-A	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-A
Zoo	59.73	55.47(12)	57.13(13)	54.12(7)	60.18(12)	**63.37(15)**	74.28	72.23(12)	71.19(13)	73.31(7)	69.92(12)	**80.01(15)**
Bcc	77.67	74.69(4)	67.77(5)	69.96(9)	73.90(5)	**80.74(6)**	79.57	74.19(4)	**81.76(5)**	74.87(9)	69.77(5)	81.31(6)
Wine	90.21	65.36(2)	96.08(11)	91.78(10)	98.15(12)	**99.35(12)**	70.56	67.72(2)	71.83(11)	72.44(10)	75.28(12)	**79.54(12)**
Hill	66.53	59.75(36)	58.83(44)	**72.10(39)**	69.57(44)	71.15(45)	81.73	62.76(36)	80.45(44)	82.57(39)	73.58(44)	**83.99(45)**
Abalone	92.16	77.35(3)	74.18(2)	88.45(3)	81.47(1)	**95.29(2)**	75.29	74.98(3)	73.78(2)	77.97(3)	84.25(1)	**87.73(2)**
Codon	71.07	69.50(11)	69.97(13)	72.09(23)	75.81(14)	**79.69(8)**	77.63	72.89(11)	71.01(13)	70.18(23)	68.80(14)	**81.57(8)**
Bean	74.18	66.32(5)	73.28(4)	78.95(5)	79.24(6)	**84.32(6)**	74.87	69.31(5)	74.11(4)	76.88(5)	75.64(6)	**79.63(6)**
Eye	76.29	73.46(10)	77.28(8)	81.11(8)	79.99(5)	**82.33(6)**	69.92	59.37(10)	58.82(8)	63.35(8)	65.81(5)	**70.01(6)**
Letter	73.99	67.89(8)	72.39(9)	77.71(8)	76.58(5)	**82.11(6)**	73.19	66.81(8)	73.79(9)	75.21(8)	74.86(5)	**80.09(6)**
Average	73.08	67.68	71.88	76.25	77.21	**82.04**	75.23	68.54	72.97	74.09	73.10	**80.43**

5.1.3. Algorithm performance evaluation

In multi-label classification, we often use two indicators Evaluation of classification learning algorithm, namely Average Precision(AP), Ranking Loss(RL).

Let the test set be $\boldsymbol{Z} = \{(x_i, Y_i)\}_{i=1}^{n} \subset R^d \times \{+1, -1\}^q$, according to prediction function $f_t(x)$ sorting functions can be defined as $\text{rank}(x, l) \in \{1, 2, \ldots, q\}$.

Average Precision(AP): The average precision (AP) is used to investigate the probability that the marker ranked in front of the sample marker in the ranking of all samples still belongs to the sample marker. The larger the value, the better the performance of the algorithm is defined as

$$\text{avgPre}(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|R_i|}\sum_{l\in R_i}\frac{\left\{k \mid \text{ran}\,k_f\,(x_i, k) \leqslant \text{ran}\,k_f\,(x_i, l)\,,\, k \in R_i\right\}}{\text{ran}\,k_f\,(x_i, l)} \tag{24}$$

Ranking Loss(RL): The average probability that irrelevant tags of all samples are ranked before relevant tags. The smaller the value is, the better the algorithm performance is

$$\text{rLoss}(f) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|R_i|\left|\overline{R_i}\right|} \cdot \left|\left\{(l, k) \mid \text{ran}\,k_f\,(x_i, l) \geqslant \text{ran}\,k_f\,(x_i, k)\,,\, (l, k) \in R_i \times \overline{\text{R}}_\text{i}\right\}\right| \tag{25}$$

This part uses four different classifiers BayesNet, Random forest, Knn and Adaboost to conduct experiments. Tables 8 and 9 list the experimental results of HAR-A algorithm and the other four algorithms on two evaluation indexes on nine data sets (take the mean value of the effect of the four classifiers). For AP evaluation index, the larger the value, the better the algorithm performance. For RL evaluation index, the smaller the value, the better the algorithm performance.

It can be seen from the above results that the performance of HAR-A algorithm is superior to others.

5.1.4. Summary

Through the comparative experiment on the algorithm from the effectiveness, efficiency and performance evaluation, it can be concluded that the HAR-A algorithm proposed by us is superior to others. The computation time required by HAR-A algorithm to obtain feasible reduction is much shorter than the other algorithms, and the results obtained are more accurate.

5.2. Performance evaluation of HAR-D algorithm

At this part, we analyze algorithm HAR-D from classification accuracy, algorithm efficiency and index performance evaluation. The specific details are as follows.

5.2.1. Algorithm classification accuracy comparison

At this part, the classification accuracy of the HAR-D algorithm proposed in this paper is compared with the other four algorithms. From every data set in Table 5, 50% features are randomly selected as the raw feature set, and the left 50% will be the deleted features. Algorithms HAR-D, HAR, DRSQR, FEAR, and NRSAR are used to compute fresh reductions while as the left 50% features are deleted from the raw 50% feature set. The experimental results are shown in Tables 10 and 11, where "raw" stands for the classification accuracy of the raw attribute set.

As shown in the above chart, the classification accuracy of algorithm HAR-D is almost higher than that of other algorithms in all cases, and its average score is far ahead, so the classification accuracy of HRA-D algorithm is very high.

5.2.2. Algorithm efficiency comparison

At this part, we test the efficiency of the algorithm HAR-D and compare it with the other four algorithms in terms of calculation time and acceleration ratio. For each data set in Table 5, five test sets were built. First, 50% of the features are randomly selected as the raw feature set. We then randomly delete features from the remaining 50% to the raw feature set to get a dynamic data set to test (that is, randomly select 10%, 20%, 30%, 40%, and 50% of the

Table 8
Performance comparison of algorithms under AP evaluation index.

Datasets	NRSAR	FEAR	DRAQR	HAR	HAR-A
Zoo	0.4215 ± 0.0124	0.4175 ± 0.0172	0.4342 ± 0.0287	0.4177 ± 0.0219	**0.5709 ± 0.0351**
Bcc	0.5928 ± 0.0213	0.5130 ± 0.0197	0.5213 ± 0.0211	0.3267 ± 0.0325	**0.6001 ± 0.0112**
Wine	0.5327 ± 0.0184	0.5243 ± 0.0145	0.4769 ± 0.0231	0.5627 ± 0.0190	**0.5797 ± 0.0107**
Hill	0.4355 ± 0.0148	0.5155 ± 0.0324	0.6294 ± 0.0342	0.3522 ± 0.0134	**0.6318 ± 0.0214**
Abalone	0.3927 ± 0.0356	0.3468 ± 0.0557	0.2213 ± 0.0562	0.7388 ± 0.0192	**0.7422 ± 0.0031**
Codon	0.4147 ± 0.0426	0.3927 ± 0.0233	0.3527 ± 0.0477	0.2419 ± 0.0334	**0.5122 ± 0.0123**
Bean	0.5318 ± 0.0156	0.4133 ± 0.0143	0.4462 ± 0.0354	0.1785 ± 0.0270	**0.5466 ± 0.0115**
Eye	0.3436 ± 0.0142	0.4435 ± 0.0119	0.4178 ± 0.0385	0.4656 ± 0.0287	**0.4772 ± 0.0174**
Letter	0.4005 ± 0.0133	0.5112 ± 0.0344	0.5231 ± 0.0441	0.7211 ± 0.0291	**0.7388 ± 0.0196**
Average	0.4518	0.4531	0.4470	0.4450	**0.5999**

Table 9
Performance comparison of algorithms under RL evaluation index.

Datasets	NRSAR	FEAR	DRAQR	HAR	HAR-A
Zoo	0.6232 ± 0.2406	0.5937 ± 0.2130	0.5931 ± 0.2981	0.6208 ± 0.2736	**0.5542 ± 0.3173**
Bcc	0.2628 ± 0.4424	0.2617 ± 0.4151	0.2638 ± 0.4500	0.2583 ± 0.4075	**0.2369 ± 0.3709**
Wine	0.4909 ± 0.3428	0.4676 ± 0.2873	0.4734 ± 0.4204	0.4892 ± 0.3466	**0.4341 ± 0.2797**
Hill	0.4516 ± 0.3539	0.4151 ± 0.3673	0.4106 ± 0.3763	0.4472 ± 0.3210	**0.3894 ± 0.4090**
Abalone	0.4033 ± 0.2694	0.3834 ± 0.2305	0.3953 ± 0.2451	0.3993 ± 0.2656	**0.3488 ± 0.2280**
Codon	0.5738 ± 0.2754	0.5398 ± 0.2473	0.5571 ± 0.2826	0.5656 ± 0.3558	**0.5132 ± 0.2700**
Bean	0.4015 ± 0.3606	0.3779 ± 0.3347	0.3835 ± 0.3462	0.3973 ± 0.3395	**0.3423 ± 0.3446**
Eye	0.7626 ± 0.4820	0.7503 ± 0.4508	0.7494 ± 0.3829	0.7528 ± 0.4514	**0.7159 ± 0.4871**
Letter	0.4917 ± 0.3243	0.4665 ± 0.3927	0.4854 ± 0.2452	0.4958 ± 0.2351	**0.4355 ± 0.1752**
Average	0.4957	0.4729	0.4791	0.4918	**0.4411**

Table 10
The comparison of classification accuracies of different algorithms on Bayes Net and Random Tree (%).

Datasets	BayesNet						RandomTree					
	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-D	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-D
Zoo	80.63	59.74(10)	69.87(12)	79.27(5)	80.18(11)	**82.33(13)**	70.56	71.18(10)	73.38(12)	62.94(5)	63.77(11)	**74.17(13)**
Bcc	88.97	79.88(4)	76.54(4)	85.67(8)	86.62(5)	**89.47(6)**	84.48	79.77(4)	82.21(4)	**85.79(8)**	75.92(5)	73.39(6)
Wine	82.97	77.39(2)	79.55(9)	84.22(8)	85.88(10)	**86.94(10)**	85.59	74.98(2)	63.97(9)	87.53(8)	81.17(10)	**88.04(10)**
Hill	79.28	76.27(26)	81.44(28)	**84.42(35)**	77.22(38)	78.95(37)	65.72	62.19(26)	66.28(28)	65.91(35)	71.19(38)	**78.88(37)**
Abalone	91.13	87.65(3)	86.78(2)	88.19(2)	83.37(1)	**92.01(2)**	70.56	72.20(3)	74.75(2)	66.98(2)	64.78(1)	**75.99(2)**
Codon	84.76	75.62(21)	73.64(17)	82.57(13)	81.90(15)	**84.82(9)**	74.72	70.79(21)	68.97(17)	70.45(13)	71.47(15)	**79.28(9)**
Bean	84.81	76.23(4)	81.92(5)	81.93(5)	82.29(8)	**85.22(8)**	70.99	70.54(4)	70.18(5)	68.32(5)	65.44(8)	**72.13(8)**
Eye	84.16	78.54(7)	75.53(8)	81.56(6)	82.05(3)	**85.38(4)**	58.24	57.73(7)	57.64(8)	50.59(6)	52.94(3)	**60.11(4)**
Letter	83.99	77.89(6)	79.31(7)	87.17(3)	83.46(5)	**87.51(9)**	71.88	67.71(6)	69.98(7)	71.11(3)	67.78(5)	**73.89(9)**
Average	84.52	76.58	78.29	83.89	82.55	**85.85**	72.53	69.68	69.71	69.96	68.23	**75.10**

Table 11
The comparison of classification accuracies of different algorithms on Knn and Adaboost (%).

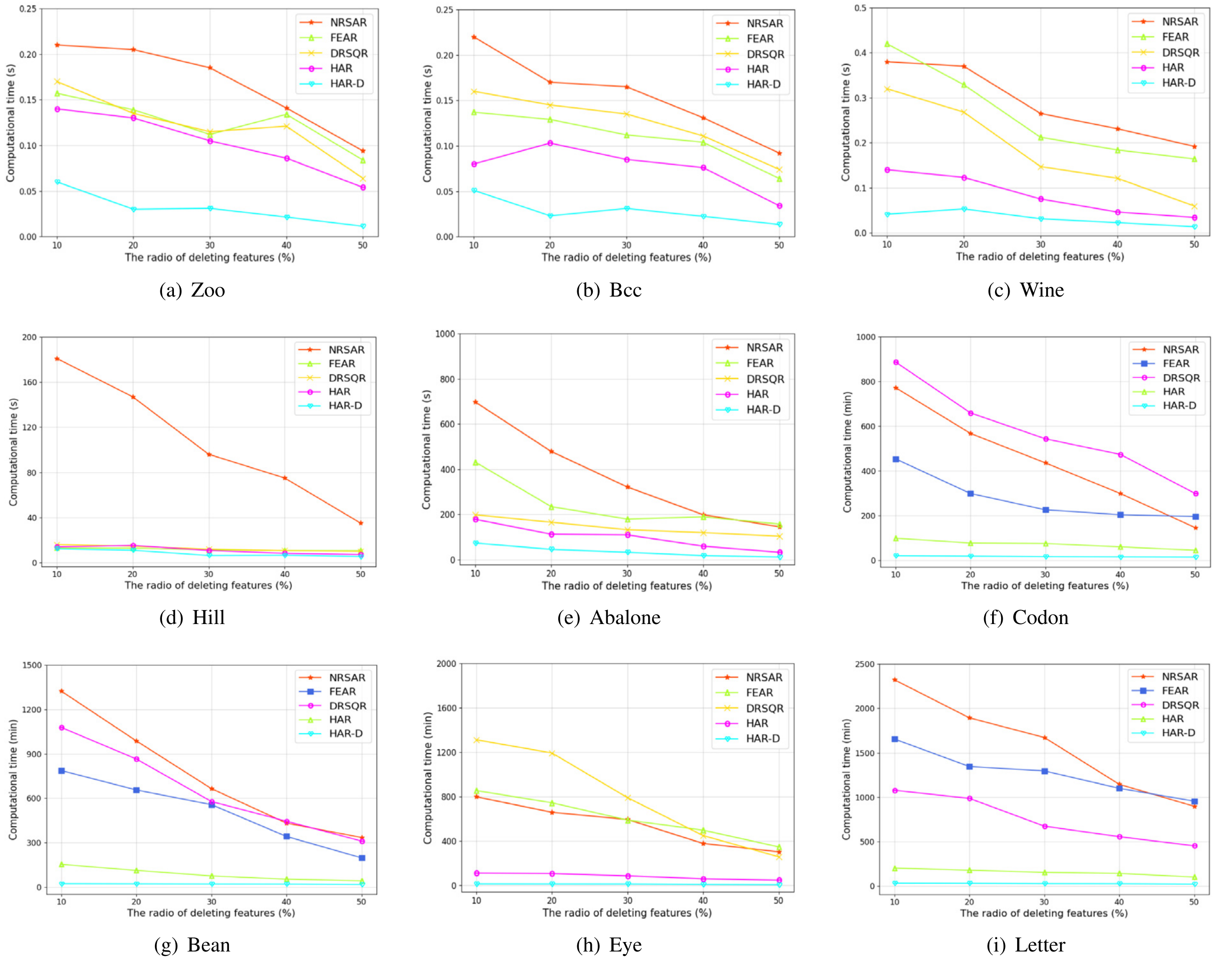
Datasets	Knn						Adaboost					
	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-D	Raw	NRSAR	FEAR	DRAQR	HAR	HAR-D
Zoo	64.57	63.28(10)	65.51(12)	70.93(5)	69.91(11)	**72.37(13)**	77.19	72.67(10)	74.93(12)	69.96(5)	71.15(11)	**77.85(13)**
Bcc	82.23	81.11(4)	79.56(4)	78.87(8)	81.87(5)	**83.54(6)**	80.01	**84.41(4)**	82.64(4)	83.38(8)	82.74(5)	83.99(6)
Wine	64.80	65.24(2)	63.68(9)	59.97(8)	65.51(10)	**66.86(10)**	77.69	62.65(2)	71.44(9)	73.98(8)	80.06(10)	**83.55(10)**
Hill	84.61	**87.23(26)**	82.58(28)	83.37(35)	85.59(38)	86.01(37)	82.13	77.98(26)	79.81(28)	80.67(35)	82.54(38)	**84.76(37)**
Abalone	65.36	67.88(3)	64.89(2)	70.13(2)	64.97(1)	**72.11(2)**	75.62	77.64(3)	69.94(2)	72.79(2)	74.98(1)	**76.85(2)**
Codon	72.65	70.79(21)	71.29(17)	73.85(13)	71.78(15)	**75.43(9)**	77.27	75.83(21)	72.55(17)	73.02(13)	73.42(15)	**80.01(9)**
Bean	72.99	76.81(4)	73.35(5)	71.14(5)	76.01(8)	**78.98(8)**	75.65	73.41(4)	76.88(5)	77.23(5)	79.23(8)	**79.75(8)**
Eye	73.34	76.46(7)	74.95(8)	68.38(6)	80.08(3)	**81.39(4)**	71.65	62.76(7)	70.54(8)	70.78(6)	68.79(3)	**72.77(4)**
Letter	71.83	72.99(6)	70.94(7)	70.83(3)	75.77(5)	**76.74(9)**	79.22	69.82(6)	75.21(7)	75.53(3)	77.48(5)	**80.12(9)**
Average	72.49	73.53	71.86	71.94	74.52	**77.05**	77.38	73.02	74.88	75.26	76.71	**79.96**

remaining 50% features and delete them from the original feature set). The time spent using different algorithms on these data sets is then compared. Fig. 4 shows the detailed variation trend of these five algorithms when the characteristics of different data sets change. The abscissa stands for the size of the feature set deleted, and the ordinate stands for the computation time.

We can see from Fig. 4 that the computation time of these five algorithms will decrease with the constant decrease of attribute feature set. It can be seen from each subgraph that the computation time of algorithm HAR-D is significantly less than that of other algorithms. Especially for large data sets, the algorithm HAR-D has a very obvious time-saving effect. Therefore, we can conclude that the efficiency of algorithm HAR-D is very high.

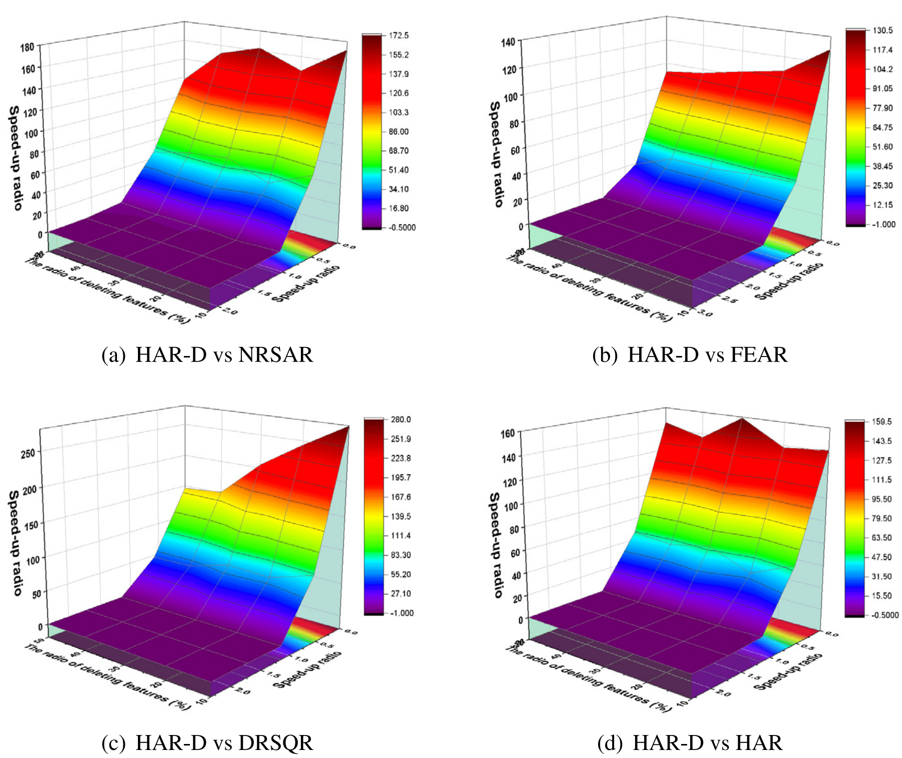
Then, we prove the validity of the algorithm HAR-D again from the perspective of acceleration ratio. Based on the results shown in Fig. 4, we calculated the acceleration ratio of the algorithm HAR-D compared with the other four algorithms. The experimental results are shown in Fig. 5.

As can be seen from Fig. 5, the acceleration ratio of algorithm HAR-D against other algorithms in all data sets is more than 0. This indicates that algorithm HAR-D is faster than the other four algorithms on all experimental data sets. Besides, for relatively



(a) Zoo (b) Bcc (c) Wine
(d) Hill (e) Abalone (f) Codon
(g) Bean (h) Eye (i) Letter

Fig. 4. The computational time of different algorithms versus different ratios of deleting features.



(a) HAR-D vs NRSAR (b) HAR-D vs FEAR
(c) HAR-D vs DRSQR (d) HAR-D vs HAR

Fig. 5. The speed-up ratios that algorithm HAR-D relates to different algorithms.

Table 12
Performance comparison of algorithms under AP evaluation index.

Datasets	NRSAR	FEAR	DRAQR	HAR	HAR-D
Zoo	0.7617 ± 0.2890	0.6920 ± 0.0322	0.7027 ± 0.0298	0.7403 ± 0.0270	**0.7715 ± 0.0329**
Bcc	0.1363 ± 0.0264	0.1367 ± 0.0260	0.1363 ± 0.0264	0.1353 ± 0.0253	**0.1411 ± 0.0218**
Wine	0.4797 ± 0.0220	0.4633 ± 0.0275	0.4607 ± 0.0321	0.4740 ± 0.0271	**0.4845 ± 0.0252**
Hill	0.6810 ± 0.0381	0.6433 ± 0.0382	0.6217 ± 0.0391	0.6733 ± 0.0313	**0.6911 ± 0.0309**
Abalone	0.4960 ± 0.0241	0.4637 ± 0.0209	0.4723 ± 0.0437	0.4883 ± 0.0192	**0.4997 ± 0.0204**
Codon	0.8103 ± 0.0219	0.7713 ± 0.0314	0.7783 ± 0.0259	0.7970 ± 0.0321	**0.8276 ± 0.0294**
Bean	0.5307 ± 0.0487	0.5143 ± 0.0495	0.5150 ± 0.0456	0.5303 ± 0.0469	**0.5331 ± 0.0397**
Eye	0.7557 ± 0.0220	0.7367 ± 0.0198	0.7490 ± 0.0286	0.7610 ± 0.0241	**0.7755 ± 0.0431**
Letter	0.4726 ± 0.1216	0.4511 ± 0.0237	0.4168 ± 0.0356	0.4223 ± 0.0145	**0.4889 ± 0.0227**
Average	0.5693	0.5414	0.5392	0.5580	**0.5792**

Table 13
Performance comparison of algorithms under RL evaluation index.

Datasets	NRSAR	FEAR	DRAQR	HAR	HAR-D
Zoo	0.0627 ± 0.0008	0.0612 ± 0.0010	0.0620 ± 0.0010	0.0621 ± 0.0008	**0.0601 ± 0.0014**
Bcc	0.0287 ± 0.0030	0.0287 ± 0.0029	0.0287 ± 0.0029	0.0285 ± 0.0030	**0.0281 ± 0.0028**
Wine	0.0442 ± 0.0024	0.0418 ± 0.0021	0.0426 ± 0.0029	0.0441 ± 0.0025	**0.0396 ± 0.0024**
Hill	0.0442 ± 0.0012	0.0436 ± 0.0014	0.0426 ± 0.0011	0.0442 ± 0.0012	**0.0418 ± 0.0016**
Abalone	0.0508 ± 0.0011	0.0478 ± 0.0015	0.0488 ± 0.0019	0.0504 ± 0.0012	**0.0443 ± 0.0016**
Codon	0.0653 ± 0.0025	0.0647 ± 0.0024	0.0641 ± 0.0025	0.0649 ± 0.0026	**0.0638 ± 0.0023**
Bean	0.0363 ± 0.0013	0.0334 ± 0.0015	0.0356 ± 0.0018	0.0357 ± 0.0010	**0.0306 ± 0.0014**
Eye	0.0357 ± 0.0009	0.0356 ± 0.0009	0.0357 ± 0.0009	0.0356 ± 0.0009	**0.0350 ± 0.0008**
Letter	0.0264 ± 0.0001	0.0287 ± 0.0013	0.0267 ± 0.0003	0.0457 ± 0.0015	**0.0190 ± 0.0011**
Average	0.0438	0.0428	0.0430	0.0457	**0.0403**

large data sets, algorithm HAR-D is tens or even hundreds of times faster than the other four algorithms. The above proves again that the efficiency of algorithm HAR-D is very high.

5.2.3. Algorithm performance evaluation

The experimental principle in the previous section. Tables 12 and 13 list the experimental results of HAR-D algorithm and other four algorithms on two evaluation indexes in nine data sets (take the average effect of the four classifiers). For AP evaluation index, the larger the value, the better the algorithm performance. For RL evaluation index, the smaller the value, the better the algorithm performance.

It can be seen from the above results that the performance of algorithm HAR-D is superior to others.

5.2.4. Summary

Through the comparative experiment on the algorithm from the effectiveness, efficiency and performance evaluation, it can be concluded that the HAR-D algorithm proposed by us is superior to others. The computation time required by HAR-D algorithm to obtain feasible reduction is much shorter than the other algorithms, and the results obtained are more accurate.

6. Summary and future research direction

In this paper, dynamic attribute feature reduction algorithm is proposed. First of all, it introduces some basic knowledge of attribute reduction. Then the related concepts of dominance relation matrix and dominance conditional entropy are introduced. Subsequently, two feature incremental attribute reduction algorithms HAR-A and HAR-D are proposed. Finally, experiments are carried out to demonstrate the accuracy, efficiency and excellent performance of the proposed algorithm.

Changes in ozone-depleting substances are likely to be multi-faceted. Applying dynamic attribute reduction algorithm to more complex dynamic data environment is a very meaningful research direction, worthy of further study. To be specific, our future research work mainly has three aspects. (1) For the change of the number of objects in the data set, we will develop an incremental attribute reduction algorithm. (2) The dynamic attribute reduction algorithm is applied to the dominant fuzzy rough set model. (3) We will further study the incremental attribute reduction method of set-valued decision information system.

CRediT authorship contribution statement

Weihua Xu: Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Validation. **Yifei Yang:** Data curation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing.

Declaration of competing interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Nos. 62376229,61976245).

References

[1] Yaojin Lin, Qinghua Hu, Jinghua Liu, Jinjin Li, Xindong Wu, Streaming feature selection for multilabel learning based on fuzzy mutual information, IEEE Trans. Fuzzy Syst. 25 (6) (2017) 1491–1507.

[2] Qinghua Hu, Lingjun Zhang, Yucan Zhou, Witold Pedrycz, Large-scale multimodality attribute reduction with multi-kernel fuzzy rough sets, IEEE Trans. Fuzzy Syst. 26 (1) (2018) 226–238.

[3] Yaojin Lin, Yuwen Li, Chenxi Wang, Jinkun Chen, Attribute reduction for multi-label learning with fuzzy rough set, Knowl.-Based Syst. 152 (2018) 51–61.

[4] Anhui Tan, Weizhi Wu, Yuhua Qian, Jiye Liang, Jinkun Chen, Jinjin Li, Intuitionistic fuzzy rough set-based granular structures and attribute subset selection, IEEE Trans. Fuzzy Syst. 27 (3) (2019) 527–539.

[5] Degang Chen, Yanyan Yang, Ze Dong, An incremental algorithm for attribute reduction with variable precision rough sets, Appl. Soft Comput. 45 (2016) 129–149.

[6] Yunge Jing, Tianrui Li, Chuan Luo, Shijinn Horng, Guoyin Wang, Zeng Yu, An incremental approach for attribute reduction based on knowledge granularity, Knowl.-Based Syst. 104 (2016) 24–38.

[7] Yanyan Yang, Degang Chen, Wang Hui, Active sample selection based incremental algorithm for attribute reduction with rough sets, IEEE Trans. Fuzzy Syst. 25 (4) (2017) 825–838.

[8] Guangming Lang, Mingjie Cai, Hamido Fujita, Qimei Xiao, Related families-based attribute reduction of dynamic covering decision information systems, Knowl.-Based Syst. 162 (2018) 161–173.

[9] Wenhao Shu, Wenbin Qian, Yonghong Xie, Incremental approaches for feature selection from dynamic data with the variation of multiple objects, Knowl.-Based Syst. 163 (2019) 320–331.

[10] Y. LeCun, B. Boser, J.S. Denker, D. Henderson, R.E. Howard, W. Hubbard, L.D. Jackel, Backpropagation applied to handwritten zip code recognition, Neural Comput. 1 (4) (1989) 541–551.

[11] G.E. Hinton, S. Osindero, Y.W. Yee, A fast learning algorithm for deep belief nets, Neural Comput. 18 (7) (2006) 1527–1554.

[12] R.J. Williams, D. Zipser, A learning algorithm for continually running fully recurrent neural networks, Neural Comput. 1 (2) (1989) 270–280.

[13] Lei Zhao, Qing Hua Hu, Wen Wu Wang, Heterogeneous feature selection with multi-modal deep neural networks and sparse group lasso, IEEE Trans. Multimed. 17 (11) (2015) 1936–1948.

[14] Vijay Bhaskar Semwal, Kaushik Mondal, Gora Chand Nandi, Robust and accurate feature selection for humanoid push recovery and classification: deep learning approach, Neural Comput. Appl. 28 (3) (2017) 565–574.

[15] Jingxia Chen, Zijing Mao, Ru Zheng, Yufei Huang, Lifeng He, Feature selection of deep learning models for EEG-based RSVP target detection, IEICE Trans. Inf. Syst. 102-D (4) (2019) 836–844.

[16] Tong Niu, Jianzhou Wang, Haiyan Lu, Wendong Yang, Pei Du, Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting, Expert Syst. Appl. 148 (2020) 1–17.

[17] Qin Zou, Lihao Ni, Tong Zhang, Qian Wang, Deep learning based feature selection for remote sensing scene classification, IEEE Geosci. Remote Sens. Lett. 12 (11) (2015) 2321–2325.

[18] Hongtao Shi, Hongping Li, Dan Zhang, Chaqiu Cheng, Xuanxuan Cao, An efficient feature generation approach based on deep learning and feature selection techniques for traffic classification, Comput. Netw. 132 (2018) 81–98.

[19] Weiping Ding, Chinteng Lin, Witold Pedrycz, Multiple relevant feature ensemble selection based on multilayer co-evolutionary consensus mapreduce, IEEE Trans. Cybern. 50 (2) (2020) 425–439.

[20] Kaustuv Nag, Nikhil R. Pal, Feature extraction and selection for parsimonious classifiers with multiobjective genetic programming, IEEE Trans. Evol. Comput. 24 (3) (2020) 454–466.

[21] Mahdieh Labani, Parham Moradi, Mahdi Jalili, A multi-objective genetic algorithm for text feature selection using the relative discriminative criterion, Expert Syst. Appl. 149 (2020) 1–21.

[22] Jianbin Ma, Xiaoying Gao, A filter-based feature construction and feature selection approach for classification using genetic programming, Knowl.-Based Syst. 196 (2020) 1–14.

[23] Asit K. Das, Sunanda Das, Arka Ghosh, Ensemble feature selection using bi-objective genetic algorithm, Knowl.-Based Syst. 123 (2017) 116–127.

[24] Anda Li, Bing Xue, Mengjie Zhang, Multi-objective feature selection using hybridization of a genetic algorithm and direct multisearch for key quality characteristic selection, Inform. Sci. 523 (2020) 245–265.

[25] Lin Sun, Xiaoyu Zhang, Yuhua Qian, Jiucheng Xu, Shiguang Zhang, Feature selection using neighborhood entropy-based uncertainty measures for gene expression data classification, Inform. Sci. 502 (2019) 18–41.

[26] Jianhua Dai, Qinghua Hu, Hu Hu, Debiao Huang, Neighbor inconsistent pair selection for attribute reduction by rough set approach, IEEE Trans. Fuzzy Syst. 26 (2) (2018) 937–950.

[27] Changzhong Wang, Yan Wang, Mingwen Shao, Yuhua Qian, Degang Chen, Fuzzy rough attribute reduction for categorical data, IEEE Trans. Fuzzy Syst. 28 (5) (2020) 818–830.

[28] Zdzislaw Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 11 (5) (1982) 341–356.

[29] Greco, Matarazzo, Slowinski, Rough approximation of a preference relation by dominance relations, Eur. J. Oper. Res. 117 (1) (1999) 63–83.

[30] Salvatore Greco, Benedetto Matarazzo, Roman Slowinski, Rough sets theory for multicriteria decision analysis, Eur. J. Oper. Res. 129 (1) (2001) 1–47.

[31] Jerzy Blaszczyski, Salvatore Greco, Roman Slowiski, Marcin Szelg, Monotonic variable consistency rough set approaches, Internat. J. Approx. Reason. 50 (7) (2009) 979–999.

[32] Wojciech Kotlowski, Krzysztof Dembczyski, Salvatore Greco, Roman Slowiski, Stochastic dominance-based rough set model for ordinal classification, Inform. Sci. 178 (21) (2008) 4019–4037.

[33] Abbas Ali, Muhammad Irfan Ali, Noor Rehman, Soft dominance based rough sets with applications in information systems, Internat. J. Approx. Reason. 113 (2019) 171–195.

[34] Xiaoxia Zhang, Degang Chen, E.C.C. Tsang, Generalized dominance rough set models for the dominance intuitionistic fuzzy information systems, Inform. Sci. 378 (2017) 1–25.

[35] Jiye Liang, Feng Wang, Chuangyin Dang, Yuhua Qian, A group incremental approach to feature selection applying rough set technique, IEEE Trans. Knowl. Data Eng. 26 (2014) 294–308.

[36] Xiao Zhang, Changlin Mei, Degang Chen, Yanyan Yang, Jinhai Li, Active incremental feature selection using a fuzzy rough set-based information entropy, IEEE Trans. Fuzzy Syst. 28 (5) (2020) 901–915.

[37] Yanyan Yang, Degang Chen, Wang Hui, Xizhao Wang, Incremental perspective for feature selection based on fuzzy rough sets, IEEE Trans. Fuzzy Syst. 26 (3) (2018) 1257–1273.

[38] Yanyan Yang, Shiji Song, Degang Chen, Xiao Zhang, Discernible neighborhood counting based incremental feature selection for heterogeneous data, Int. J. Mach. Learn. Cybern. 11 (5) (2020) 1115–1127.

[39] Wenhao Shu, Wenbin Qian, Yonghong Xie, Incremental feature selection for dynamic hybrid data using neighborhood rough set, Knowl.-Based Syst. 194 (2020) 1–15.

[40] Ye Liu, Lidi Zheng, Yeliang Xiu, Hong Yin, Suyun Zhao, Xizhao Wang, Hong Chen, Cuiping Li, Discernibility matrix based incremental feature selection on fused decision tables, Internat. J. Approx. Reason. 118 (2020) 1–26.

[41] Asit K. Das, Shampa Sengupta, Siddhartha Bhattacharyya, A group incremental feature selection for classification using rough set theory based genetic algorithm, Appl. Soft Comput. 65 (2018) 400–411.

[42] Degang Chen, Lianjie Dong, Jisheng Mi, Incremental mechanism of attribute reduction based on discernible relations for dynamically increasing attribute, Soft Comput. 24 (1) (2020) 321–332.

[43] Feng Wang, Jiye Liang, Yuhua Qian, Attribute reduction: A dimension incremental strategy, Knowl.-Based Syst. 39 (2013) 95–108.

[44] Guangming Lang, Duoqian Miao, Mingjie Cai, Zhifei Zhang, Incremental approaches for updating reducts in dynamic covering information systems, Knowl.-Based Syst. 134 (2017) 85–104.

[45] Anping Zeng, Tianrui Li, Dun Liu, Junbo Zhang, Hongmei Chen, A fuzzy rough set approach for incremental feature selection on hybrid information systems, Fuzzy Sets and Systems 258 (2015) 39–60.

[46] Feng Wang, Jiye Liang, Chuangyin Dang, Attribute reduction for dynamic data sets, Appl. Soft Comput. 13 (1) (2013) 676–689.

[47] Wei Wei, Xiaoying Wu, Jiye Liang, Junbiao Cui, Yijun Sun, Discernibility matrix based incremental attribute reduction for dynamic data, Knowl.-Based Syst. 140 (2018) 142–157.

[48] Wei Wei, Peng Song, Jiye Liang, Xiaoying Wu, Accelerating incremental attribute reduction algorithm by compacting a decision table, Int. J. Mach. Learn. Cybern. 10 (9) (2019) 2355–2373.

[49] Mingjie Cai, Guangming Lang, Hamido Fujita, Zhenyu Li, Tian Yang, Incremental approaches to updating reducts under dynamic covering granularity, Knowl.-Based Syst. 172 (2019) 130–140.

[50] Lianjie Dong, Degang Chen, Incremental attribute reduction with rough set for dynamic datasets with simultaneously increasing samples and attributes, Int. J. Mach. Learn. Cybern. 11 (6) (2020) 1339–1355.

[51] Yunge Jing, Tianrui Li, Hamido Fujita, Baoli Wang, Ni Cheng, An incremental attribute reduction method for dynamic data mining, Inform. Sci. 465 (2018) 202–218.

[52] Claude Shannon, Warren Weaver, The mathematical theory of communication, Bell Syst. Tech. J. 27 (3/4) (1948) 373–423.

[53] Qinghua Hu, Xunjian Che, Lei Zhang, David Zhang, Maozu Guo, Daren Yu, Rank entropy based decision trees for monotonic classification, IEEE Trans. Knowl. Data Eng. 24 (11) (2012) 2052–2064.