Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 6876144, 16 pages
https://doi.org/10.1155/2022/6876144

Research Article

Indistinguishable Element-Pair Attribute Reduction and Its Incremental Approach

Baohua Liang, [1,2,3] **Haiqi Zhang**, [1,2] **Zhengyu Lu**, [1,2] **and Zhengjin Zhang** [3]

[1] *Institute of Computer Science and Engineering, Guangxi Normal University, Guilin 541004, Guangxi, China*
[2] *Guangxi Collaborative Innovation Center of Multi-source Information Integration and Intelligent Processing, Guangxi Normal University, Guilin 541004, China*
[3] *School of Information Engineering, Chaohu University, Hefei 238000, Anhui, China*

Correspondence should be addressed to Zhengjin Zhang; 054029@chu.edu.cn

Received 24 May 2022; Accepted 8 August 2022; Published 29 September 2022

Academic Editor: Jerzy Baranowski

Attribute reduction is a popular approach of preprocessing data. Discernibility matrix is a typical method that focuses on attribute reduction. Faced with the processing of modern information systems with large amounts of data and rapid changes, the traditional static discernibility matrix reduction model is powerless. To overcome this shortcoming, this paper first proposes an indistinguishable element pair method that does not need to store discernibility information, which retains the advantages of institution and easy-to-understand, and at the same time effectively solves the problem of space consumption. In order to make the model adapt to the processing of dynamic data sets, we further study the incremental mechanism and design a set of dynamic reduction models, which can adjust the reduction set in time according to the changes of objects. Theoretical analysis and experimental results indicate that the proposed algorithm is obviously superior to the discernibility matrix model, and can effectively deal with the reduction of dynamic data sets.

1. Introduction

Rough sets theory (RST) is a valid mathematical tool, which was proposed by Pawlak and Skowron in 1982, for dealing with inaccurate, incomplete, and vague information [1]. RST has been widely used in many fields such as machine learning [2], data mining [3], decision supporting [4], expert system [5], pattern recognition [6], and music emotions annotation [7]. Attribute reduction is one of the hot research focuses in RST [8], which aims to delete redundant data, while keeping the distinguishing power of the original data in information systems. For the convenience of the following description, Table 1 summarizes the list of abbreviations in the article. In the last two decades, many heuristic attribute reduction approaches have been developed based on the positive region [9], discernibility matrix [10, 11], information entropy [12], fuzzy rough [13, 14], m-polar fuzzy [15, 16], and knowledge granularity [17].

Among the abovementioned approaches, DMA is a typical reduction model. Since DMA consumes a lot of space to store distinguishable information, it cannot reduce large data sets. In order to effectively express the distinguishable information among samples, Hu and Cercone [18] proposed a concise definition of a discernibility matrix. Ye and Chen [19] proposed a discernibility matrix-elements that retains all bases of 1. Yang and Sun [20] use the sample comparison of the upper and lower approximation to obtain the discernibility matrix. Dong et al. [21] proposed a fast algorithm of attribute reduction for covering the decision system with minimal elements in discernibility matrix. Wei et al. [22] proposed two discernibility matrices in the sense of entropies. However, these approaches only consider how to improve the distinguishing ability of samples, and do not consider the space consumption. In order to reduce the space computation, Jiang [10] proposed a minimal element selection tree. Li et al. [23] proposed a simple object-attribute discernibility matrix approach. Although scholars have improved the discernibility matrix, the space consumption problem has not been fundamentally solved. To overcome

Table 1: The list of abbreviations.

Abbreviation	Original
RST	Rough sets theory
DMA	Discernibility matrix approach
IEP	Indistinguishable element-pair
IEPAO	Indistinguishable element-pair adding objects
IEPDO	Indistinguishable element-pair deleting objects
CPNR	Compute the positive and negative region

this deficiency, this paper proposes a method based on IEP without discernibility matrix. Firstly, we divide the data set according to conditional attributes and decision attributes and calculate the number of indistinguishable element pairs. Then, select the conditional attribute with the smallest values of IEP. Finally, repeat the abovementioned two steps until the value is 0.

With the rapid development of communication and network techniques, the actual data may change over time. However, the IEP method is only suitable for static data sets. Hence, it is desired to design an incremental attribute reduction algorithm with IEP to deal with dynamic decision systems.

Incremental learning is an efficient approach making full use of the precious results of the original decision system, which can obtain the efficient reduced results by recomputing the updated part of the dynamic data set. Many incremental algorithms have been proposed with different models for dynamic data. Yang proposed an incremental algorithm for updating an object or attribute [24]. Ge et al. developed an incremental attribute reduction based on a simplified discernibility matrix, which is equivalent to attribute reduction based on a positive region [25]. Liu et al. proposed a strong discernibility matrix method for incremental attribute reduction on fuzzy decision tables [26]. In literature [27], Wei proposed three new types of discernibility matrices by compacting a decision table. Zhang et al. proposed a method based on a relation matrix under the change attribute reduction in set-valued information systems [28]. Ma et al. [29] proposed a compressed binary discernibility matrix to process the group dynamic data. Obviously, the abovementioned matrix methods mainly focus on updating the elements of discernibility matrix. These approaches are ineffective in obtaining the reduction results with large-scale decision systems due to the limited memory space. Hence, we incorporate the incremental update mechanism into the IEP approach. Verifies the feasibility and efficiency of proposed algorithm through extensive experiments on UCI data sets.

2. Preliminaries

In this section, we review some basic concepts about rough set, discernibility matrix, and indistinguishable element-pair.

2.1. Basic Concepts

Definition 1 (see [1]). Given the decision system is a quadruple tuple $S = (U, A, V, f)$, where U is a finite nonempty object set and A is a finite nonempty attribute set, $V = \cup_{a\in A} V_a, V_a$ is a set of its values, and $f\colon U \times A \longrightarrow V$ is an information function with $f(x, a) = V_a$ for each $a \in A$ and $x \in U$. If $A = C \cup D$, where C is the conditional attribute set, and D is the decision attribute set. For every subset $P \subseteq A$, an indiscernibility relation $\mathrm{IND}(P)$ is defined as follows:

$$\mathrm{IND}(P) = \{(x, y) \in U \times U \mid \forall a \in P, f(x, a) = f(y, a)\}. \quad (1)$$

Obviously, if $\mathrm{IND}(P)$ denotes as U/P, U/P is an equivalence relation. We assume includes x, the equivalence relation x is defined as:

$$[x]_P = \{y \mid \forall a \in P, f(x, a) = f(y, a)\}. \quad (2)$$

Definition 2 (see [1]). Given the decision system $S = (U, A, V, f)$ for every subset $Y \subseteq U$ and indiscernibility relation $\mathrm{IND}(P)$, the upper approximation set and the lower approximation set of Y can be defined by the basic set of P as follows:

$$\begin{aligned} \underline{P}(Y) &= \{x \in U \mid [x]_P \subseteq Y\}, \\ \overline{P}(Y) &= \left\{x \in U \middle| [x]_P \cap Y \neq \varnothing)\right\}. \end{aligned} \quad (3)$$

The universe U is partitioned into three disjoint regions by these two approximations $\overline{P}(Y)$ and $\underline{P}(Y)$: the positive region $\mathrm{POS}_P(Y)$, the negative region $\underline{P}(Y)$, and the boundary region $\mathrm{BND}_P(Y)$. Then the three different regions are defined as following, respectively:

$$\begin{aligned} \mathrm{NEG}_P(Y) &= U - \overline{P}(Y) \\ \mathrm{BND}_P(Y) &= \overline{P}(Y) - \underline{P}(Y) \\ \mathrm{POS}_P(Y) &= \underline{P}(Y). \end{aligned} \quad (4)$$

Definition 3 (see [18]). Let $DT = (U, C \cup D)$ be a decision table, C be the condition attribute set, and D be the decision attribute. The discernibility matrix in all samples is defined as $M_{DT}^{D} = \{mD/ij\}$, where:

$$m_{ij}^{D} = \begin{cases} \{c \in C\colon f(x_i, c) \neq f(x_j, c)\}, & f(x_i, d) \neq f(x_j, d), \\ \varnothing, & \text{otherwise}. \end{cases} \quad (5)$$

Definition 4 (see [31]). Let $DT = (U, C \cup D)$ be a decision table, C be the condition attribute set, and D be the decision attribute. In terms of a positive region, the discernibility matrix is defined as $M_{DT}^{P} = \{m_{ij}^{P}\}$,where:

$$m_{ij}^{P}=\begin{cases}\left\{c\in C:\ f\left(x_i,c\right)\neq f\left(x_j,c\right)\right\}, & f\left(x_i,d\right)\neq f\left(x_j,d\right)\text{and}x_i,x_j\in U_1,\\ \left\{c\in C:\ f\left(x_i,c\right)\neq f\left(x_j,c\right)\right\}, & x_i\in U_1,x_j\in U_2,\\ \varnothing, & \text{otherwise.}\end{cases}\tag{6}$$

U_1 is the consistent part of the decision table and U_2 is the inconsistent part DT.

Definition 5 (see [22]). Let $DT=(U,C\cup D)$ be a decision table, C be the condition attribute set, and D be the decision attribute. The discernibility matrix in the sense of complement entropy is defined as $ME/DT=\{mE/ij\}$.,where:

$$m_{ij}^{E}=\begin{cases}\left\{c\in C:\ f\left(x_i,c\right)\neq f\left(x_j,c\right)\right\}, & f\left(x_i,d\right)\neq f\left(x_j,d\right)\text{ and }x_i,x_j\in U_1\\ \left\{c\in C:\ f\left(x_i,c\right)\neq f\left(x_j,c\right)\right\}, & x_i\in U_1,x_j\in U_2\\ \left\{c\in C:\ f\left(x_i,c\right)\neq f\left(x_j,c\right)\right\}, & x_i,x_j\in U_2\\ \varnothing, & \text{otherwise}\end{cases}\tag{7}$$

U_1 and U_2 are the same as Definition 4.

Theorem 1. *In the discernibility matrix information in Definition 5, the total number of pairwise comparisons between elements is only related to the division of decision attributes of the data objects.*

Proof. Suppose U is a nonempty finite set of data objects. U_1 is the consistent part of U which belongs to the set of positive regions. U_2 is the inconsistent part of U, which is included in the negative region set. The samples in U_1 and U_2 are not duplicated. Let $[U/D]=\left\{D_1,D_2,\ldots,D_n,D_{\text{neg}}\right\},D_{\text{POS}}=D_1\cup D_2\cup\ldots\cup D_n$, then $U_1=D_{\text{pos}}$, and $U_2=D_{\text{neg}}$.

Suppose $|\cdot|$ is the cardinality of the data set, we can have $|U|=|D_{\text{pos}}|+|D_{\text{neg}}|$. According to Definition 5, it can be seen that there are three cases for comparison between samples.

Firstly, the positive region samples with different values of decision attribute should be compared in pairs where $x_i,x_j\in U_1$ and $f(x_i,d)\neq f(x_j,d)$. Due to the symmetry of $f(x_i,d)\neq f(x_j,d)$ and $f(x_j,d)\neq f(x_i,d)$, the sample x_i and x_j are compared twice, the actual number of comparisons should be halved. Let count1 be the number of comparisons between samples of the positive region set, then we can achieve the following conclusion:

$$\text{count1}=\frac{\left[|D_1|\cdot\left(\left|D_{\text{pos}}\right|-|D_1|\right)+|D_2|\cdot\left(\left|D_{\text{pos}}\right|-|D_2|\right)+\ldots+|D_n|\cdot\left(\left|D_{\text{pos}}\right|-|D_n|\right)\right]}{2=\left(\left|D_{\text{pos}}\right|^2-\sum_{i=1}^{n}|D_i|^2\right)/2}.\tag{8}$$

Secondly, let count2 be the number of comparisons between samples of the positive region and negative region, then we have $\text{count2}=|D_{\text{pos}}|\cdot|D_{\text{neg}}|$ where $x_i\in U_1,x_j\in U_2$.

Thirdly, all samples among the negative region are compared with other samples, and repeated comparisons should be subtracted, we have $\text{count3}=|D_{\text{neg}}|\cdot(|D_{\text{neg}}|-1)/2$.

Overall, the total number (TotalCount) of comparisons in discernibility matrix based on Definition 5 is as follows:

$$\text{Totalcount} = \text{Count1} + \text{Count2} + \text{Count3} = \frac{\left(\left|D_{\text{pos}}\right|^2 - \sum_{i=1}^{n}\left|D_i\right|^2\right)}{2} + \left|D_{\text{pos}}\right| \cdot \left|D_{\text{neg}}\right| + \left|D_{\text{neg}}\right|$$
$$\cdot \frac{\left(\left|D_{\text{neg}}\right| - 1\right)}{2} = \frac{\left[|U|^2 - |U| - \sum_{i=1}^{n}\left(\left|D_i\right|^2 - \left|D_i\right|\right)\right]}{2}$$
$$= C_{|U|}^2 - \sum_{i=1}^{n} C_{\left|D_i\right|}^2 = \frac{\left[|U|^2 - |U| - \sum_{i=1}^{n}\left(\left|D_i\right|^2 - \left|D_i\right|\right)\right]}{2}$$
$$= C_{|U|}^2 - \sum_{i=1}^{n} C_{\left|D_i\right|}^2. \tag{9}$$

Obviously, when a data set is given, the total number of comparisons is only related to the division of decision attributes, and irrelevant to conditional attributes. □

2.2. The Presentation of the Indistinguishable Element-Pair. The discernibility matrix algorithm records the differences between samples by different values of conditional attributes between them, and the amount of distinguishable information measures the importance of the attributes. The larger the value, the more important the attribute. For the discernibility matrix proposed in literature [24] and literature [22], the number of comparisons between samples in the discernibility matrix is determined when the data set is given. Among the samples to be compared, the value of certain condition attributes is either the same or different. The same values of one conditional attribute mean being indistinguishable, while different means being distinguishable. If the amount of distinguishable information is larger, the amount of indistinguishable information is smaller when the total number of comparisons does not change. Here, we use the amount of indistinguishable information to measure the importance of conditional attributes.

Definition 6. Suppose U is a universe that is nonempty finite data set, A is a conditional attribute set. $U/A = \{X_1, X_2, \cdots X_n\}$ is the division of data set U on conditional attribute A and $U/D = \{Y_1, Y_2, \cdots, Y_m, Y_{\text{neg}}\}$ is the division on decision attribute D. Y_{neg} is the division of inconsistent samples in decision attributes The indistinguishable element-pair of A relative to D is defined as follows:

$$IEP_U(D|A) = \sum_{i=1}^{n}\left(C_{\left|X_i\right|}^2 - \sum_{k=1}^{m} C_{\left|X_i \cap Y_k\right|}^2\right), \tag{10}$$

where $C_{\left|X_i\right|}^2 = |X_i| \cdot (|X_i| - 1)/2$.

In fact, all data objects among the subdivision X_i are indistinguishable from each other. There are $C_{\left|X_i\right|}^2$ pairs. However, among these element pairs, some comparisons should be subtracted due to some data objects with the same decision attribute value. We have the definition of indistinguishable element-pair.

The asterisked $(*)$ data objects belong to the negative region set

Example 1. Suppose U is a simplified decision table without repeated samples in Table 2. A is a conditional attribute and D is a decision attribute. Let $A = a \cup b$, the data objects $\{x_9, x_{10}\}$ belong to the negative region, Let $U/D = \{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6\}, \{x_7, x_8\}, \{x_9, x_{10}\}\}$. and $U/A = \{\{\mathbf{1}, \mathbf{2}, \mathbf{5}, \mathbf{6}\}, \{3, 4, 9\}, \{\mathbf{7}, \mathbf{8}, 10\}\}$.In subdivisions U/A, the underlined data objects have the same decision attribute value. Data set U has three subdivisions $U_1 = \{\mathbf{1}, \mathbf{2}, \mathbf{5}, \mathbf{6}\}, U_2 = \{3, 4, 9\}$ and $U_3 = \{\mathbf{7}, \mathbf{8}, 10\}$ on conditional attribute A. According to Definition 6, we have indistinguishable information of the three subdivisions as follows:

$$\begin{aligned}
IEP_{U_1}(D|A) &= C_4^2 - C_2^2 - C_2^2,\\
IEP_{U_2}(D|A) &= C_3^2,\\
IEP_{U_3}(D|A) &= C_3^2 - C_2^2,\\
IEP_U(D|A) &= IEP_{U_1}(D|A) + IEP_{U_2}(D|A) + IEP_{U_3}(D|A) = 9.
\end{aligned} \tag{11}$$

Theorem 2. *The smaller the indistinguishable element pair, the stronger the distinguishing ability.*

Proof. According to Theorem 1, if the data set is given, the number of data objects in the positive and negative regions is

Table 2: Simplified decision table.

No	a	b	c	D
1	1	0	0	1
2	1	2	0	1
3	1	2	0	1
4	1	2	1	2
5	1	0	2	2
6	1	0	3	2
7	0	0	1	3
8	0	0	2	3
9	1	2	3	*
10	0	0	3	*

also definite. Obviously, according to Definition 5, the number of comparisons between all data objects in the data set is also an invariable number. Suppose TotalCount is the total number of comparisons and the indistinguishable element-pair based on conditional attribute A is $\mathrm{IEP}_U(D|A)$, then the discernibility element-pair's TotalCount-$\mathrm{IEP}_U(D|A)$. So, the smaller the $\mathrm{IEP}_U(D|A)$, the bigger the TotalCount-−$\mathrm{IEP}_U(D|A)$, the stronger the discernibility. □

Theorem 3. *Given the decision system $S=(U,C\cup D)$ and $B\subseteq P\subseteq C$. Then $U/(D|P)$ is detailed of $U/(D|B)$, and we have $IEP_U(D|P)\leq IEP_U(D|B)$*

Proof. Let $U/B=\{x_1,x_2,\cdots,x_j\cup x_{j+1},x_{j+2},\cdots,x_n\}$ and $U/P=\{x_1,x_2,\cdots,x_j,x_{j+1},\cdots,x_n\}$. Suppose $x_j\cup x_{j+1}$ is a subdivision of U/B. The detail subdivision based on P is $\{x_j\}$ and $\{x_{j+1}\}$ and $x_j\cap x_{j+1}=\varnothing$, the other subdivision is unchanged. Let $U/D=\{Y_1,Y_2,\cdots,Y_m\}$. For convenience below, let $\Delta=\sum_{i=1\wedge i\neq j\wedge i\neq j+1}^{n}(C^2_{|X_i|}-\sum_{k=1}^{m}C^2_{|X_i\cup Y_k|})$, $|X_j|=x$, $|X_{j+1}|=y$, $\sum_{k=1}^{m}|X_j\cap Y_k|=ax$, $\sum_{k=1}^{m}|X_{j+1}\cap Y_k|=by$ and $0\leq a,b\leq 1$, we have the result as follows:

$$
\begin{aligned}
IEP_U(D|B) &= \Delta + C^2_{|X_j\cup X_{j+1}|}-\sum_{k=1}^{m}C^2_{|(X_j\cup X_{j+1})\cap Y_k|}\\
&= \Delta+\frac{(x+y)(x+y-1)}{2}-\frac{(ax+by)(ax+by-1)}{2}IEP_U(D|P)\\
&= \Delta + C^2_{|x_j|}+C^2_{|x_{j+1}|}\\
&\quad -\sum_{k=1}^{m}C^2_{|x_j\cap Y_k|}-\sum_{k=1}^{m}C^2_{|x_{j+1}\cap Y_k|}=\Delta+\frac{x(x-1)}{2}+\frac{y(y-1)}{2}-\frac{ax(ax-1)}{2}\\
&\quad -\frac{by(by-1)}{2}IEP_U(D|B)-IEP_U(D|P)=xy-axby.
\end{aligned}
\tag{12}
$$

Since $0\leq a,b\leq 1$, $\mathrm{IEP}_U(D|B)-\mathrm{IEP}_U(D|P)=C^2_{|x_i\cup x_j|}$

$$-C^2_{|x_i|}-C^2_{|x_j|}\geq 0,\text{Weave }\mathrm{IEP}_U(D|P)\leq\mathrm{IEP}_U(D|B) \tag{13}$$

□

Theorem 4. *Let $S=(U,C\cup D)$ be a decision table and U is a data set without duplicate samples, then $IEP_U(D|C)=0$.*

Proof. Assume $U/C=\{U_1,U_2,\ldots,U_n\}$, Since $U\prime$ is a data set without duplicate samples, $|U_1|=|U_2|=\ldots=|U_n|=1$, So $\mathrm{IEP}_U(D|C)=\sum_{i=1}^{n}0$. □

Definition 7. Let $S=(U,C\cup D)$ be a decision table and $B\subseteq C$. U is a data set without duplicate samples. Then B is a relative reduction based on the following indistinguishable element-pair of S if B satisfies:

(1); $\mathrm{IEP}_U(D|B)=\mathrm{IEP}_U(D|C)$

(2); $\forall a\in B,\mathrm{IEP}_U(D|(B-\{a\}))\neq\mathrm{IEP}_U(D|B)$

Definition 8 (see [30]). Let $S=(U,A)$ be an information system. For any $a\in A$ the value of object x about attribute a is $f(a,x)$. Let $\mathrm{Dis}(a)=\{(x_i,x_j)|f(a,x_i)\neq f(a,x_j)\}$ and $\mathrm{Dis}(A)=\cup_{a\in A}\mathrm{Dis}(a)$, we called Dis(a) and Dis(A) the discernibility relations in terms of a and A, respectively.

Definition 9 (see [30]). Suppose $U=\{x_1,x_2,\cdots,x_n\}$, let $M_S=(c_{ij})_{n\times n}$ denote a $n\times n$ matrix, where $c_{ij}=\{a\in A\colon f(a,x_i)\neq f(a,x_j)\}$ for $(x_i,x_j)\in\mathrm{Dis}(A)$, otherwise $c_{ij}=\varnothing$, M_S is called the discernibility matrix of the information system $S=(U,A)$.

3. The Algorithm Based on Indistinguishable Element-Pair

The indistinguishable element-pair algorithm obtains the importance of attributes by means of discernibility matrix information and does not create the discernibility matrix. We should compute the positive region and negative region at first, then achieve the simplified decision table. Reduce the calculation of duplicate data objects, saving a lot of time.

3.1. Compute the Positive and Negative Region (CPNR). In each detailed subdivision, if the decision value of the sample is different, then we put the first sample x into the negative region set and let $U_{\mathrm{neg}}=U_{\mathrm{neg}}$ and $\cup x$ the rest put the x into the positive region set and let $U = U_{\text{pos}} \cup U_{\text{neg}}$. In the process of calculating positive and negative regions, equivalence class division needs to be calculated continuously. Here is an ingenious method, which can greatly speed up the calculation speed of equivalence class partitioning. The details are described as follows: for i in range (n): list [array[i] AppendixAppendix for clarity. (i).

If the data object has an integer value on the attribute, the characteristics of an integer can be used. By collecting all the objects with the same value in the same subdivision, equivalence class division can be obtained quickly and accurately.

Example 2. There are six data objects, the values of attribute A are 1, 2, 1, 3, 2, and 1, respectively. Let Array [1–6] = {1, 2, 1, 3, 2, 1}, collecting the data objects with the same value into the same list. Array [1] = Array [3] = Array [6] = 1, we have list [1] = {1, 3, 6}. Array [2] = Array [5] = 2, we have list [2] = {2, 5}. Array [4] = 3, we have list [3] = {4}.

Abovementioned all, data objects 1, 3, and 6 are divided into the same subdivision, and data objects 2 and 5 are divided into the same subdivision. (Algorithm 1)

3.2. The Attribute Reduction Algorithm Based on Indistinguishable Element-Pair (IEP). Suppose $f(x, a)$ is the value of the data object x on the conditional attribute A. There have two different data objects x_i and x_j, if $f(a, x_i) \neq f(a, x_j)$, then a is recorded in the discernibility matrix. Another way to think about it is to take down indistinguishable data objects. After research, all samples divided by the same subdivision are indistinguishable. Algorithm IEP is described as follows: (Algorithm 2)

Example 3. Suppose $U\prime$ is a simplified decision table without repeated samples in Table 3. Based on the definition equivalence class, we have $U\prime/a \cup D = \{\{1, 4\}, \{\mathbf{2}, \mathbf{3}\}, \{5, 6^*\}\}$.The bold data objects of the subdivision have the same value on the attribute $a \cup D$. The asterisked ($*$) data objects belong to the negative region set. Based on the Definition 6, we have $\text{IEP}_{U\prime}(D|a) = C_2^2 + 0 + C_2^2 = 2$, $U\prime/b \cup D = \{\{\mathbf{1}, \mathbf{2}, \mathbf{3}, 4\}, \{5, 6^*\}\}$, $\text{IEP}_{U\prime}(D|b) = C_4^2 - C_3^2 + C_2^2 = 4$, $U\prime/c \cup D = \{\{\mathbf{1}, \mathbf{3}, 6^*\}, \{2, 5\}, \{4\}\}$, $\text{IEP}_{U\prime}(D|c) = C_3^2 - C_2^2 + C_2^2 + 0 = 3$, $U\prime/e \cup D = \{\{\mathbf{1}, \mathbf{3}, 5, 6^*\}, \{2, 4\}\}$ $\text{IEP}_{U\prime}(D|e) = C_4^2 - C_2^2 + C_2^2 = 6$

When the conditional attribute with the lower indistinguishable degree has a stronger distinguishing ability, we select the attribute a, and let red = $\{a\}$. If the amount of information is not zero, we enter the next cycle. On the basis of it is divided by conditional attributes {b, c, e}, we obtain the following: $U\prime/\text{red} \cup b \cup D = \{\{1, 4\}, \{2, 3\}, \{5, 6^*\}\}$ $\text{IEP}_{U\prime}(D|(\text{red} \cup b) = C_2^2 + 0 + C_2^2 = 2$, $U\prime/\text{red} \cup c \cup D = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}\}$, $\text{IEP}_{U\prime}(D|\text{red} \cup c) = 0$, $U\prime/\text{red} \cup e \cup D = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5, 6^*\}\}$ $\text{IEP}_{U\prime}$, $(D|e) = 0 + 0 + 0 + 0 + C_2^2 = 1$.

Because $\text{IEP}_{U\prime}(D|\text{red} \cup c) = 0$ is the smallest, we select c to merge into the reduced set red. Now, the amount of information is 0, and the algorithm terminates. Reduce result is red = $\{a, c\}$.

3.3. The Existing Static Reduction Algorithms. The typical discernibility matrix algorithm and the related improved algorithms constantly revise the definition of discriminant matrix from the perspective of distinguishable data objects, leading to the inevitable consumption of a large amount of space resources to store the discernibility matrix. The phenomenon of memory overflow often occurs during the reduction of large data sets, which leads to the failure to complete the reduction task.

The IEP method does not need to store discernibility matrix and is suitable for reduction of large-scale data sets. In order to further verify the effectiveness of the IEP presented in this paper, let's analyze the complexity of time and space and other similar algorithms based on discernibility matrix. In IEP algorithm, U is a decision table. Steps 2-3 focus on calculating the simplified decision table $U\prime$ and $|U\prime| \leq |U|$. The time complexity of computing $U\prime$ is $O(|U||C|)$. The time complexity of steps 5 is $O((|C|^2 - |\text{Red}|^2)|U\prime|)$. The space complexity of data set is $O(|U||C|)$, steps 5 want space $O(|U\prime|)$. Therefore, the total time complexity of algorithm IEP is $O(|U||C|) + O((|C|^2 - |\text{Red}|^2) \cdot |U|\prime)$ and the space complexity is $O(|U||C|)$. Table 3 shows a comparison of the time and space complexity of computing the reductions by Algorithms HU [18], DDMSE [22], and MEDA [30]. From Table 4, we can obtain the time complexity of IEP is usually much smaller than the algorithms HU because of $O((|C|^2 - |\text{Red}|^2) \cdot |U|\prime) + O(|U||C|)$ is lower than $O(|U||C|) + O(|U|^2|C|^2)$. The time complexity of the algorithm MEDA is $O(|U|^2|C|^2)$. The space complexity of algorithm IEP is $O(|U||C|)$ but the space complexity of storing the discernibility matrix of HU, DDMSE is $O(|U|^2|C|)$. Therefore, the space consumption of algorithm IEP is much less than that of algorithms HU, DDMSE, and MEDA.

4. Incremental Attribute Reduction Algorithm Based on Indistinguishable Element-Pair

The abovementioned algorithm IEP only adapts to the static data set. In reality, most data sets are dynamic. The traditional static methods are ineffective. Therefore, it is necessary to study some algorithms for dynamic data sets.

4.1. An Incremental Method to Calculate Indistinguishable Element-Pair after Adding Some Objects (IEPAO). There are two kinds of data objects as to updating in data set: increase and decrease. Let's introduce the first one: to increase the data objects. When some data objects are added to data set, we only need to calculate the IEP of the updating part and obtain the amount of information with the help of the previous reduction result red. If the amount of information is zero, the updated reduction result is red. Otherwise, the added part objects will be merged with the basic data, we compute the amount of information based on detailed subdivision according to attribute set red $\cup D$.

Theorem 5. *Let $S = (U, C \cup D)$ be a decision system, $U/C \cup D = \{X_1, X_2, \cdots, X_m\}$. It is assumed that $U_{\Delta x}$ is the new data objects,*

Input: $U, R = \varnothing, \text{count} = 0, C, D$
Output: $U', U'_{\text{pos}}, U'_{\text{neg}}$
$/ * U'$ is a data set without duplicate samples, U'_{pos} is a positive region, data set and U'_{neg} is a negative region data set. $*/$
Step 1: $U' = U'_{\text{pos}} = U'_{\text{neg}} = \varnothing$
Step 2: $U'' = U$
Step 3: while (count $\leq |U|$ && $R \subseteq C$) do{
Step 3.1: for any $C_i \subset C - R$, let $R = R \cup C_i$
Step 3.2: compute $U''/R = \{U''_1, U''_2, \ldots, U''_n\}$,
Step 3.3: statistics of the subdivisions regarded as
$|U''_i| = 1$, let $U'_{\text{pos}} = U'_{\text{pos}} \cup U_{ik}$, count add 1. }
Step 4: scan the remaining subdivisions.

Algorithm 1: CPNR algorithm computes the positive and negative region method.

Input: $S = (U, C \cup D)$
Output: red
Step 1: red$\leftarrow\varnothing$, $B\leftarrow\varnothing$
Step 2: Calculate the positive region U_{pos} and negative region U_{neg} with CPNR
Stet 3: get the simplified decision table U' through Step 2
Step 4: IEP$\leftarrow$Sys.maxsize, IEP_list$\leftarrow\varnothing$
Step 5: while (IEP > 0) do {
Step5.1: $\text{IEP}(b, B, D) = \min\{\text{IEP}(a, B, D), a \in C - B$
Step 5.2: IEP_list$\leftarrow$IEP_list(b, B, D)
Step 5.3: delete all the subdivisions and the card is 1 of IEP_list
Step 5.4: $B \leftarrow B \cup \{b\}\}$
Step 6: red$\leftarrow B$ return red

Algorithm 2: The indistinguishable element-pair algorithm (IEP).

Table 3: Example of decision table.

No	a	b	c	e	D
1	0	0	1	0	1
2	1	0	0	1	1
3	1	0	1	0	1
4	0	0	2	1	2
5	2	1	0	0	2
6	2	1	1	0	*

Table 4: A comparison of time and space complexity of IEP, HU, DDMSE, and MEDA.

Algorithm	Time	Space
IEP	$O(\|U\|\|C\|) + ((\|C\|^2 - \|\text{Red}\|^2) \cdot \|U\|\prime)$	$O(\|U\|\|C\|)$
HU	$O(\|U\|\|C\|) + O(\|U\|^2\|C\|^2)$	$O(\|U\|^2\|C\|)$
DDMSE	$O(\|U\|\|C\|) + O(\|U\|^2\|C\|^2)$	$O(\|U\|^2\|C\|)$
MEDA	$O(\|U\|^2\|C\|^2)$	$O(\|U\|^2)$

$U_{\Delta x}/C \cup D = \left\{Y_1, Y_2, \cdots, Y_{m\prime}\right\}, X_i \cup Y_i = X'_i, 1 \leq i \leq k.$
According to the division of $U/C \cup D, U_{\Delta x}/C \cup D$, we have $U \cup U_{\Delta x}/C \cup D = \{X'_1, X'_2, \ldots, X'_k, X_{k+1}, X_{k+2}, \ldots, X_m, Y_{k+1}, Y_{k+2}, \ldots, Y_{m\prime}\}$. Then $\text{IEP}_{U \cup U_{\Delta x}}(D|C) = \text{IEP}_U(D|C) + \text{IEP}_{U_{\Delta x}}(D|C) + |U||U_{\Delta x}| - \sum_{i=1}^{k} |X_i||Y_i|$

Proof. See Appendix 1 for the proof process.

According to Theorem 5, the value of $\text{IEP}_U(D|C)$ is related to $\text{IEP}_{U_{\Delta x}}(D|C)$, where add data objects. We propose the algorithm IEPAO based on this characteristic. (Algorithm 3)

In Table 5, red is the reduction result before adding data, red$'$ is the final reduction result IEP can only reduce static data set the time complexity is $O(|U||C|) + O((|C|^2 - |\text{red}|^2) \cdot |U|)$ If data objects $U_{\Delta x}$ is added, the time complexity becomes $O((|U| + |U_{\Delta x}|) \cdot |C|) + O((|C|^2 - |\text{red}|^2)(|U| + |U_{\Delta x}|))$. The IEPAO algorithm uses the previous reduction results, the time complexity is $O(|U_{\Delta x}| \cdot |C|) + O(((|C| - |\text{red}|)^2 - (\text{red}' - \text{red})^2) \cdot \min\{|U_{\Delta x}|, |U|\})$. It clearly shows that the calculation time of IEPAO is less than IEP. ☐

4.2. Incremental Updating Attribute Reduction Algorithm When Delete Some Objects (IEPDO). In reality, some data will be discarded after a long time. IEPDO algorithm can reduce the deleted data objects dynamically.

Theorem 6. *Let $S = \{U, C \cup D\}$ be a decision system and $U/C \cup D = \{X_1, X_2, \cdots, X_m\}$. We assume that the deleted data object set is $U_{\Delta x}$ and $U_{\Delta x}/C \cup D = \{Y_1, Y_2, \cdots, Y_k\}$. From the definition of equivalence class divided, we have $(U - U_{\Delta x})/C \cup D = \{X'_1, X'_2, \cdots, X'_k, X_{k+1}, \cdots, X_m\}$, where $X'_i = X_i - Y_i$ (i = 1, 2, …, k). If delete the data objects $U_{\Delta x}$ from U, the indistinguishable amount of information is*

Input: $U, U/\text{red} \cup D$, red and incremental object sets $U_{\Delta x}$, where U is the simplified decision table before update.
Output: Updated reduction set red′
Step 1: Mark the negative region data objects, list$\leftarrow U/\text{red} \cup D$, $\text{red}' \leftarrow \text{red}$
Step 2: Compute $\text{IEP}_{U_{\Delta x}}(D|\text{red}')$
Step 3: We may assume that $(U/\text{red}') \cap (U_{\Delta x}/\text{red}')$ has k subdivisions and $X_i \in U, Y_i \in U_{\Delta x}$, then compute $\sum_{i=1}^{k} |X_i||Y_i|$.
Step 4: Compute $\text{Info} = \text{IEP}_U(D|\text{red}') + \text{IEP}_{U_{\Delta x}}(D|\text{red}') + |U||U_{\Delta x}| - \sum_{i=1}^{k} |X_i||Y_i|$
Step 5: If Info = 0, algorithm is terminated else
While Info ≠ 0 do {
Let list$\leftarrow U \cup U_{\Delta x}/\text{red}'$
for a in $C - \text{red}'$ {
Compute $\text{IEP}_{U \cup U_{\Delta x}}(D|\text{red}'' \cup a)$
, $a* \leftarrow \min\{\text{IEP}_{U \cup U_{\Delta x}}(D|\text{red}'' \cup a)\}$
$\text{red}' \leftarrow \text{red}' \cup a*$,
list$\leftarrow (U \cup U_{\Delta x})/\text{red}'$ }
Compute $\text{Info} = \text{IEP}_{U \cup U_{\Delta x}}(D|\text{red}')$ }
Step 6: return red′

Algorithm 3: Incremental updating algorithm based on IEP when adding some objects (IEPAO).

Table 5: The time complexity of each step of algorithm IEPAO.

Step no	Time complexity	Result
Step 1	$O(\lvert U_{\Delta x}\rvert\lvert C\rvert)$	Get $U_{\Delta x}$
Step 2	$O(\lvert U_{\Delta x}/\text{red} \cup D\rvert)$	Get $\text{IEP}_{U\Delta x}(D\vert\text{red})$
Step 3	$O(\lvert U/re\ d\rvert) + O(\lvert U_{\Delta x}/\text{red}\rvert)$	Get $(U \cup U_{\Delta x})/\text{red}$
Step 4	$O(1)$	Get info
Step 5	$O(\lvert U_{\Delta x}\rvert \bullet \lvert C\rvert) + O(((\lvert C\rvert - \lvert\text{red}\rvert)^2 - (\text{red} - \text{red})^2) \bullet \min\{\lvert U_{\Delta x}\rvert, \lvert U\rvert\})$	Get red′

$IEP_{U-U_{\Delta x}}(D|C)$, then $IEP_{U-U_{\Delta x}}(D|C) = IEP_U(D|C) + IEP_{U_{\Delta x}}(D|C) - |U||U_{\Delta x}| + \sum_{i=1}^{k} |X_i||Y_i|$.

Proof.

$$\text{IEP}_{U-U_{\Delta x}}(D|C) = C^2_{|U-U_{\Delta x}|} - \sum_{i=1}^{k} C^2_{|X_i - Y_i|} - \sum_{i=k+1}^{m} C^2_{|X_i|}$$

$$= C^2_{|U|} + C^2_{|U_{\Delta x}|} - |U||U_{\Delta x}| - \sum_{i=1}^{k}\left(C^2_{|X_i|} + C^2_{|Y_i|}\right) + \sum_{i=1}^{m}|X_i||Y_i| - \sum_{i=k+1}^{m} C^2_{|X_i|} = IEP_U(D|C) + IEP_{U_{\Delta x}}(D|C) - |U||U_{\Delta x}| + \sum_{i=1}^{k}|X_i||Y_i|. \tag{14}$$

According to Theorem 6, the value of $\text{IEP}_U(D|C)$ is related to $\text{IEP}_{U_{\Delta x}}(D|C)$ where delete data objects. We propose the algorithm IEPDO based on this characteristic. (Algorithm 4)

When some objects are deleted, we have the indistinguishable information of updated data set, which needs a small amount of computation through the previous reduced result and deleted data objects. Suppose the final reduction result is red′, usually, |red′| less than or equal to |red| and less than $|C|$. The time complexity comparison of IEPDO and IEP is provided in Table 6. Obviously, the time complexity of IEPDO is smaller than IEP. □

5. Experiment Analysis

In this section, lots of experiments are conducted on both static and dynamic data sets to verify the efficiency of the proposed attribute reduction algorithms. In the experiments, fifteen data sets are downloaded from UCI. Table 7 displays the basic information of each data set, where $|U|$ represents the number of samples, $|C|$ represents the number of conditional attributes, $|D|$ represents the number of decision classes, and Type represents the decision system is consistency (Y in short) or inconsistency (N in short), respectively. For the convenience of the following description,

Input: $U, U/\text{red} \cup D$, red and delete object sets $U_{\Delta x}$, where U is the simplified decision table before update.
Output: red$'$
Step 1: Mark the deleted data objects is $U_{\Delta x}$, list$\leftarrow U/\text{red} \cup D$, red$'\leftarrow$red
Step 2: Compute $\text{IEP}_{U_{\Delta x}}(D|\text{red}')$;
Step 3: We may assume that $(U/\text{red}') \cap (U_{\Delta x}/\text{red}')$ has k subdivisions and $X_i \in U, Y_i \in U_{\Delta x}$, then compute $\sum_{i=1}^{k} |X_i||Y_i|$
Step 4: Compute Info $= \text{IEP}_U(D|\text{red}') + \text{IEP}_{U_{\Delta x}}(D|\text{red}') - |U|\|U_{\Delta x}\| + \sum_{i=1}^{k} |X_i||Y_i|$
Step 5: Let list$\leftarrow (U - U_{\Delta x})/\text{red}'$
  for a in red$'$ {
    Compute Info $= \text{IEP}_{U-U_{\Delta x}}(D|\text{red}' - a)$,
    if Info = 0 then red$'\leftarrow$red$' - a*$ and list$\leftarrow (U - U_{\Delta x})/\text{red}'$
  else
    break;
    }
Step 6: return red$'$

Algorithm 4: Increment updating algorithm based on IEP when delete some objects (IEPDO).

Table 6: The time complexity of algorithm IEPDO and IEP.

Algorithm	Time complexity
IEP	$O((\lvert U\rvert - \lvert U_{\Delta x}\rvert)\lvert C\rvert) + O((\lvert C\rvert^2 - \lvert\text{red}\rvert^2)(\lvert U\rvert - \lvert U_{\Delta x}\rvert))$
IEPDO	$O((\lvert U\rvert - \lvert U_{\Delta x}\rvert)\lvert\text{red}\rvert) + O((\lvert\text{red}\rvert^2 - \lvert\text{red}'\rvert^2)(\lvert U\rvert - \lvert U_{\Delta x}\rvert))$

Table 7: A description of data sets.

Dataset	$\lvert U\rvert$	$\lvert C\rvert$	$\lvert D\rvert$	Consistency?
Mushroom	8124	22	2	Y
Audiology	226	70	24	Y
Blance-scale	625	4	3	Y
Breast	286	9	2	Y
Car	1728	6	4	Y
Letters	20000	17	26	Y
Ticdata2000	5822	85	2	Y
Gene	3190	60	3	Y
Nursery	12960	8	5	Y
Handwritten	5620	64	10	Y
Mass	830	6	2	N
Hepatitis	155	19	2	N
Connect-4	67557	42	3	Y
Chess kr-kp	3196	36	2	Y
Spect heart	267	22	2	N

the data set Letters recognition is abbreviated as Letters, Mammographic Mass as Mass. All the character or string features are normalized into an integer. All of the experiments have been implemented on a PC with Windows 10, Core™ i7-10710U CPU 1.10 GHz 1.61 Hz and 8 G memory. All of the algorithms are coded in python, and the used software isPyCharm Community Edition 2020.2.3 × 64 and Weak3.2.

5.1. Performance Comparison between Algorithm IEP and Other Discernibility Matrix Algorithms Based on Static Data Sets. In experiment, we consider the fifteen data sets from UCI listed in Table 7. These selected data sets are reasonably distributed, including large data sets for Letters and small data sets for Hepatitis, Audiology, consistent data sets (Gene and Mushroom, etc.), and inconsistent data sets (Mass and Spect heart). In order to show the time effect of each algorithm, we refer to the SpeedupRatio= T_{baseline}/T method proposed by literature [31], is the executing time of a typical algorithm. T_{baseline} reaches its maximum when the typical algorithm cannot perform the reduction task. Then SpeedupRatio $\in [0, \infty)$.

For the different data sets, the SpeedupRatio of IEP and the other three algorithms (Hu, DDMSE, and MEDA) is also different. Table 8 shows the SpeedupRatio of IEP, Hu, DDMSE, and MEDA. The data in bold indicates that the algorithm runs the fastest on a certain data set. In Table 8, we have the unpredictable speed of Hu, DDMSE, and MEDA on Letters and Connect-4 data sets because these two data sets are too large. The SpeedupRatios of IEP are 1.6109 and 2.4286, respectively, on data sets Audiology and Hepatitis. Since the values are greater than 1, the speed of IEP is faster than Hu on Audiology and Hepatitis. From Table 8, IEP is the fastest but the difference is not obvious among the Hu, DDMSE and MEDA based on small data sets. Overall, the SpeedupRatio is related to $|U|^2 \cdot |C|$. The smaller the $|U|^2 \cdot |C|$, the faster the speed.

Table 9 shows the reduction and performance time of the comparisons of four algorithms. In Table 9, time is measured in seconds, red is reduction and the data in bold represents the minimum reduction time of many algorithms. The IEP takes only 1.953 seconds to reduce the Mushroom data set, while DDMSE takes 230.071 seconds. The main reason is that DDMSE modifies the definition of discernibility information to improve the distinguishing ability, leading to the increasing number of compared element pairs. Then, it makes the space for storing discernibility matrix larger and larger. On data sets Letters and Connect-4, IEP can quickly and effectively obtain the reduction results, while Hu, DDMSE, and MEDA cannot complete the reduction task due to insufficient memory. On the small data set Breast and Balance-scale, IEP needs to waste 0.035 seconds, while the other three algorithms take 0.191,0.152 and 0.044 seconds respectively. For the reduction on a small data set, the time effect of IEP and other algorithms is not obvious.

Compared with other discernibility matrix algorithms, IEP has less time consumption, while getting the same reduction results based on the same data sets. Especially, the reduction effect is more obvious on large-scale data sets.

5.2. Time Comparison of IEPAO and IEP When Adding Data Objects. In the following experiments, we select nine data sets for dynamic update experiments from Table 7. For each data set, 50% of the data objects are randomly selected as the original object set, and the remaining data are randomly generated at the proportions of 10%, 20%, 30%, 40%, and 50% as incremental object sets, respectively. The incremental part is divided into 5 groups of experiments, each group is executed 10 times and computes the average time. Experimental results are outlined in Figure 1. In the experiment of IEP, time statistics do not include the calculating time of original objects.

In each subfigures of Figure 1, the x-coordinate represents the increment ratio, and the unit is the proportion of the increased part of the data in the total data. The value of the y-coordinate y is the time of computing reduction in different incremental, which are measured by seconds. In Figure 1, the curve with a five-pointed star mark shows the change in the running time of IEPAO, while the curve with a circle mark indicates the variation of IEP.

It can be seen in Figure 1 that as the size of data set expands, the time of calculating reduction will increase. The calculation time of IEPAO is much less than IEP. The main reason for this phenomenon is that when we add the new objects into the data set, IEPAO only needs to calculate the added part of the data, and then combine the previous reduction results in obtaining the changed result quickly. But, IEP can only process static data sets. When new data is added, it takes longer time to recalculate the original data and the added part. On the whole, the performance of IEPAO is relatively stable in Figure 1. With the increase of updated data, the calculation time is also increasing. But, IEPAO has an anomaly in that the calculation time decreased as the data increasing. The subfigure (f) of Figure 1 displays that IEPAO takes 4.543 seconds to reduce the Letters data set with a 20% increment, and 4.253 seconds to reduce the data with 30% increment. The main reason is that when the data increases, it accelerates the division of data on conditional attributes. Since the amount of information is zero based on subdivision of cardinality 1, these objects are constantly deleted during the reduction process to accelerate the convergence speed.

5.3. Time Comparison of IEPDO and IEP When Deleting Data Objects. The same as Section 5.2, nine data sets are selected in Table 7. Take the original data of each data sets as the basic data, and randomly select 10%, 20%, 30%, 40%, and 50% objects in the remaining data to delete respectively. Use IEP and IEPDO to reduce the updated data. Each group of data was selected to repeat the experiment for 10 times and averaged the time of 10 times. The experimental results show in Figure 2. In each subfigure of Figure 2, x-axis represents the proportion of decrement data objects, while y-axis represents the computational time. The time is measured by seconds. The curve with five-pointed star mark in Figure 2 shows the change in the running time of the IEPDO, while

As shown in Figure 2, when we delete some data objects from data set, the computational time of IEP and IEPDO will decrease accordingly. In the same computing environment, IEPDO takes less time than IEP. The IEPDO method only needs to calculate the updated objects when deleting data objects. Some updated objects when deleting data objects. Some conditional attributes with zero indistinguishable element-pair are removed from the reduction set. However, IEP takes longer time because of calculating the reduced data set from all conditional attributes. For the IEP selected, it takes 0.0338 seconds on the decreased 30% data objects of Hepatitis, while taking 0.0352 seconds to compute the deleted 40% data. It costs 1.233 seconds to calculate the decrease of 30% data of Nursery but takes 1.2451 seconds on the decrease of 40% data. Why the calculation time increases with the decreasing data is that the speed of dividing equivalence class is slowed down as it randomly selects some weak distinguish power on conditional attributes. The performance of IEPDO is relatively stable. By deleting data, the calculation time decreases. From nine subfigure of Figure 2, when we reduce the large-scale data, the time-consuming effect of IEPDO is more obvious with the decreased data objects, while the effect is not significant on small data sets, such as data sets of Chess kr-kp and Hepatitis.

5.4. Classification Accuracy Analysis of IEP, IEPAO, and IEPDO. In this section, the precision of classification is calculated on the selection of reducts obtained by the algorithms IEP, IEPAO, and IEPDO. Firstly, we take 50% of objects of each data set in nine data sets from Table 7 as the basic data set, the rest 50% data as the incremental objects

Table 8: A speedup ratio comparison of IEP, MEDA, HU, and DDMSE.

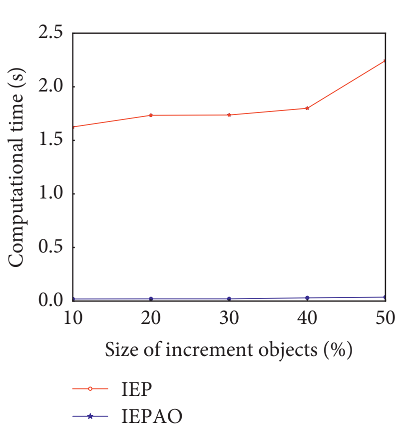
Data set	Speedup ratio			
	IEP	MEDA	HU	DDMSE
Mushroom	**81.916**	4.3723	1	0.6954
Audiology	**1.6109**	1.5839	1	0.7454
Blance-scale	**17.857**	9.1912	1	1.1220
Breast	**5.4571**	4.3409	1	1.2566
Car	**41.639**	10.5291	1	1.2320
Letters	∞	—	—	—
Ticdata2000	**19.3021**	18.1108	1	1.5578
Gene	**26.4713**	4.5756	1	0.5976
Nursery	**1397.65**	34.368	1	3.5097
Handwritten	**54.5751**	1.0171	1	0.4773
Mass	**22.3890**	13.4333	1	1.0824
Hepatitis	**2.42862**	1.8478	1	0.9884
Connect-4	∞	—	—	—
Chess kr-kp	**26.7601**	1.797	1	1.3320
Spect heart	**10.0600**	2.086	1	0.9860

The bolded values are the fastest speed ratio.

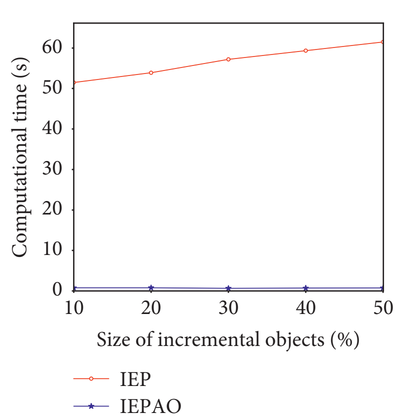
Table 9: A time and reduced comparison IEP, MEDA, HU, and DDMSE.

Data sets	IEP		MEDA		HU		DDMSE	
	|Red|	Time (s)	|Red|	Time (s)	|Red|	Time (s)	|Red|	Time (s)
Mushroom	4	**1.953**	5	36.59	4	159.981	4	230.071
Audiology	14	**0.293**	14	0.298	13	0.472	14	0.633
Lance-scale	4	**0.035**	4	0.068	4	0.625	4	0.557
Breast	9	**0.035**	9	0.044	9	0.191	9	0.152
Car	6	**0.133**	6	0.526	6	5.538	6	4.495
Letters	12	**5.892**	—	∞	—	∞	—	∞
Ticdata2000	23	**7.724**	23	8.232	23	149.085	23	95.701
Gene	10	**2.593**	10	15.001	10	68.639	10	114.863
Nursery	8	**1.32**	8	53.679	8	1844.866	8	525.649
Handwritten	7	**4.688**	8	251.558	7	255.848	7	536.041
Mass	5	**0.054**	5	0.09	5	1.209	5	1.117
Hepatitis	8	**0.039**	9	0.046	8	0.085	8	0.086
Connect-4	34	**64.358**	—	∞	—	∞	—	∞
Chess kp-kr	29	**2.22**	29	33.059	29	59.407	29	44.568
Spect heart	18	**0.035**	18	0.1685	18	0.3521	18	0.357

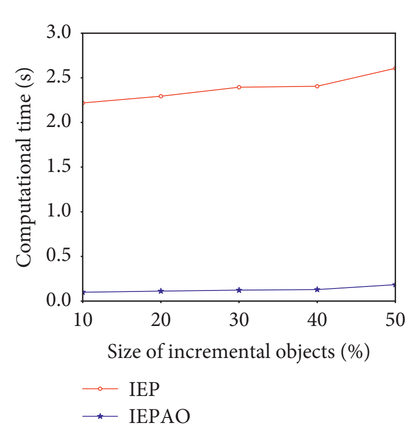
The time shown in bold is the least time.



(a)



(b)



(c)

Figure 1: Continued.

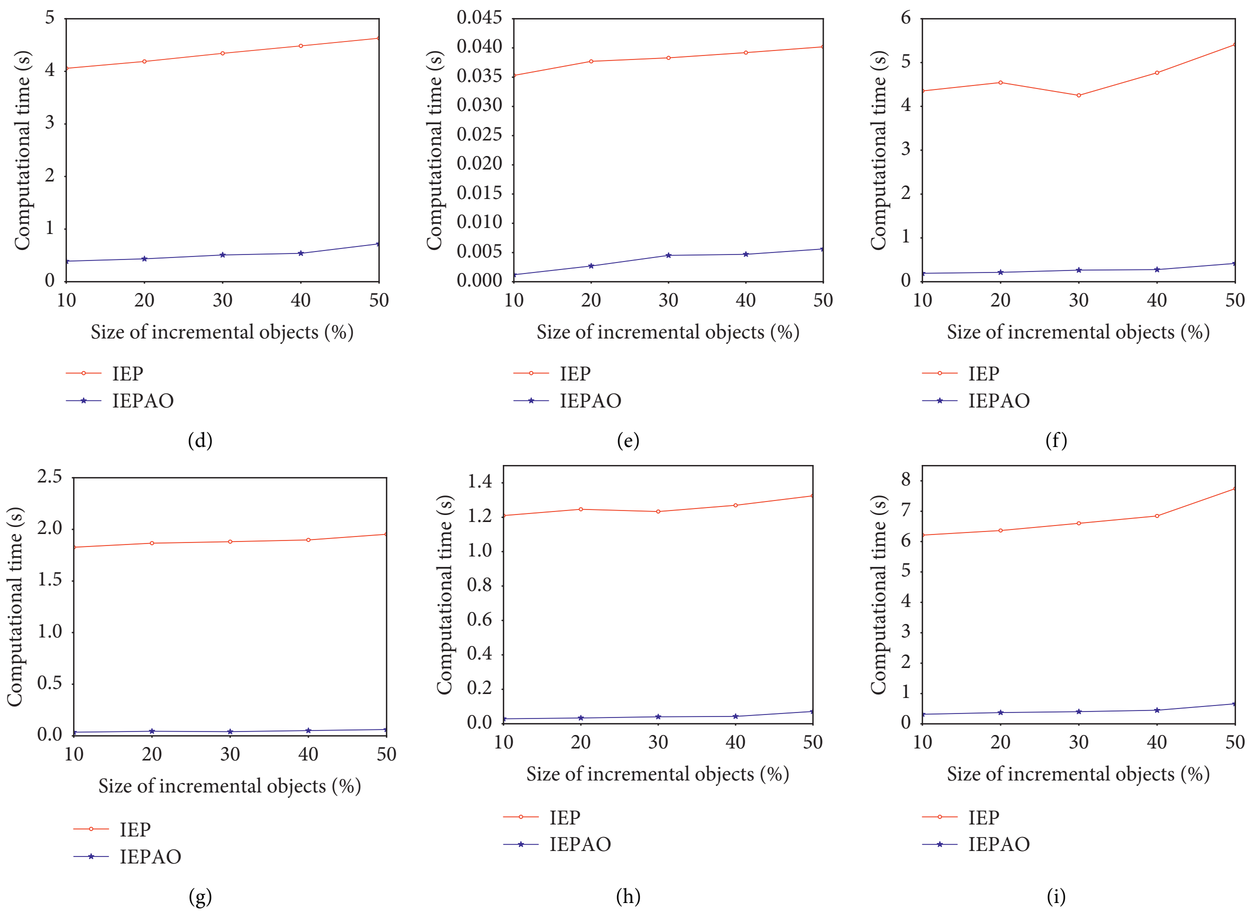


FIGURE 1: The time comparison of IEP and IEPAO when adding data objects. (a) Chess kr-kp. (b) Connect-4. (c) Gene. (d) Handwritten. (e) Hepatitis. (f) Letters. (g) Mushroom. (h) Nursery. (i) Ticdata2000.

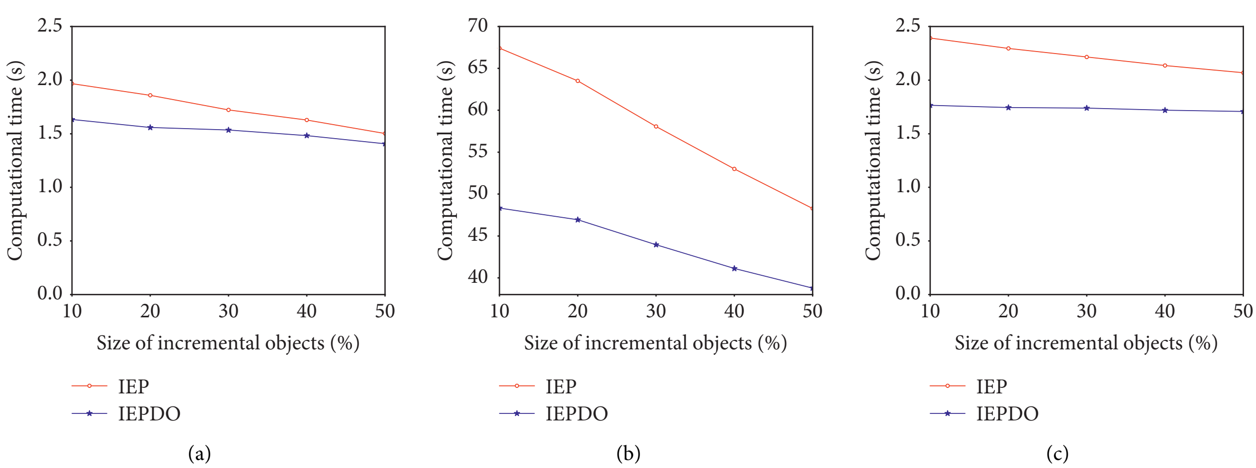


FIGURE 2: Continued.

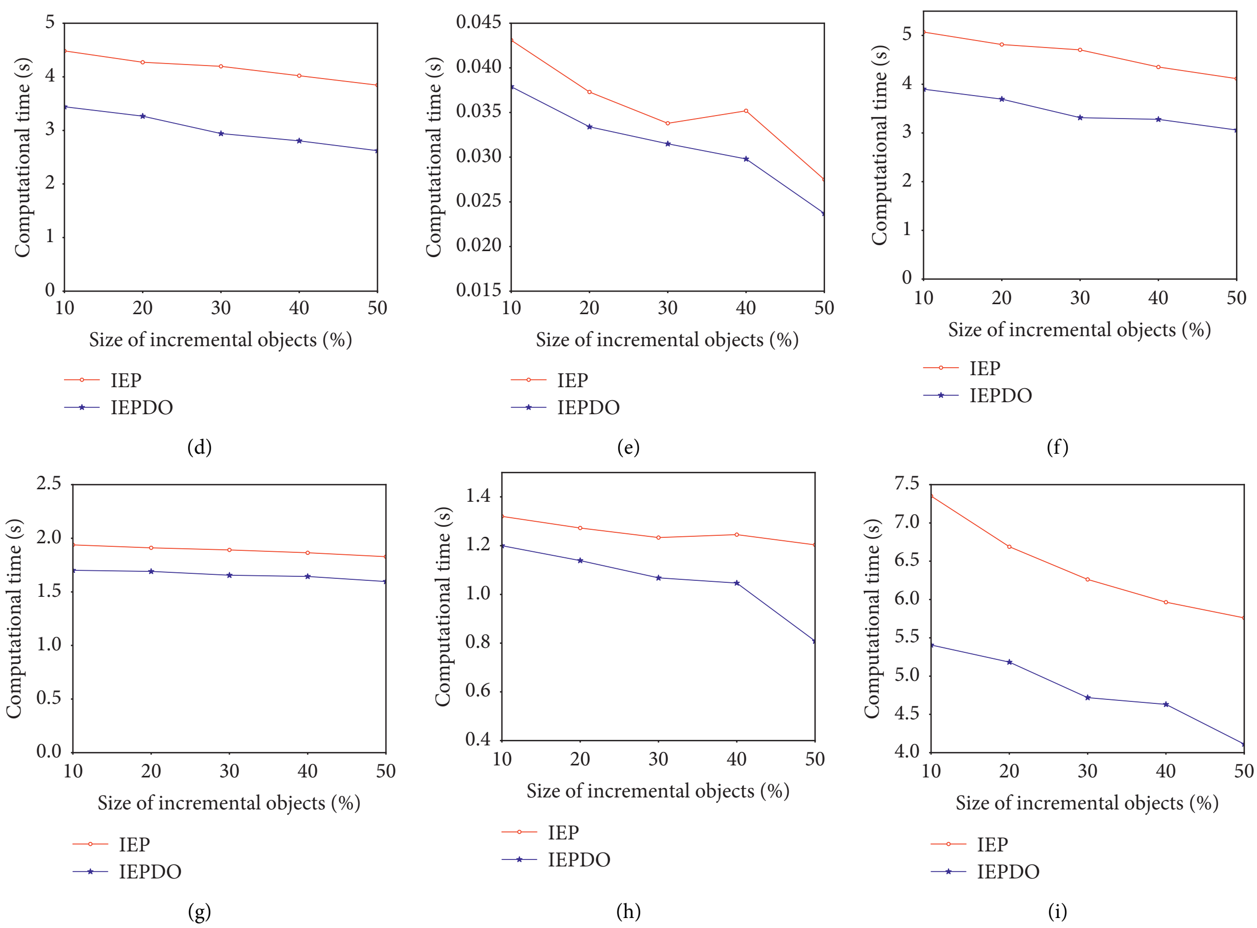


Figure 2: The time comparison of IEP and IEPDO when deleting data objects. The curve with circle mark shows the calculation time variation form of IEP. (a) Chess kr-kp. (b) Connect-4. (c) Gene. (g) Mushroom. (h) Nursery. (i) Ticdata2000.

Table 10: A comparison of IEP and IEPAO on classification accuracy.

Data set	J48		RF		NB		SMO	
	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)
Chess kr-kp	99.4368	**99.4368**	99.0926	**99.0926**	88.3292	**88.3292**	95.4318	95.4318
Connect-4	80.9006	**83.6573**	82.1099	**85.0756**	72.2027	**74.7908**	80.2002	**82.4056**
Gene	65.7053	**68.0923**	70.3762	**72.9031**	67.2424	**69.6412**	66.6771	**69.4532**
Handwritten	65.3203	**67.5629**	71.9757	**72.9084**	62.6868	**64.0985**	67.4021	**70.2341**
Hepatitis	79.3548	**82.5094**	82.5806	**84.5620**	80.0001	**83.8007**	80.0001	**83.0924**
Letters	87.5150	**89.3318**	95.2050	**95.2050**	60.2500	**62.6780**	76.6250	**79.9138**
Mushroom	100.00	100.00	100.00	100.00	98.6091	98.6091	100.00	100.00
Nursery	97.0525	**97.0525**	99.0664	99.0664	90.3241	90.3241	93.0787	93.0787
Ticdata2000	82.0225	**84.4521**	80.1498	**83.4097**	77.9026	**79.8094**	82.7157	**86.7732**

The bold values are the classification accuracy with the best classification performance.

and select the algorithms IEP and IEPAO to reduce. Secondly, we delete 50% of objects randomly from each data set, using the algorithms IEP and IEPDO to process. Then, the classification accuracies are acquired by using J48, Naive-Bayes (NB), RandomForest (RF), SMO classifier, and 10-fold cross-validation. The experimental results are shown in Tables 10 and 11.

From Table 10, it is clear that when some objects are added into the information systems, the average classification accuracy of the reduction found by incremental algorithm IEPAO is better than those of algorithm IEP in data sets Chess, Connect-4, Gene, Handwritten, Hepatitis, Letters and Ticdata2000 are coincide with those of algorithm IEP in data sets, e.g., Chess kr-kp, Mushroom, and Nursery. The experimental results show that the incremental algorithm IEPAO can find a feasible attribute reduction when incremental algorithm IEPAO replaces algorithm IEP. Moreover, the algorithm IEPAO can obtain high-quality attribute reduction with less time consumption. Similarly, when some objects are deleted from the original object set, the average classification accuracy of the reduction obtained by the algorithm IEPDO is better than

Table 11: A comparison of IEP and IEPDO on classification accuracy.

Data set	J48		RF		NB		SMO	
	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)	IEP (%)	IEPAO (%)
Chess kr-kp	99.4368	99.4368	99.2925	99.2925	99.2925	99.2925	96.7315	96.7315
Connect-4	81.6038	**84.4270**	83.3241	**86.0821**	73.4528	**77.5632**	81.2032	**84.3158**
Gene	67.3467	**69.9341**	72.3478	**75.9318**	68.2463	**71.3484**	68.8741	**72.5629**
Handwritten	66.8630	**69.4582**	72.9360	**74.9348**	63.6868	**65.5376**	68.8053	**74.6523**
Hepatitis	81.3372	**84.9823**	84.2367	**86.5892**	81.0206	**84.9341**	82.5690	**85.6538**
Letters	88.8510	**91.4426**	96.0027	**96.8721**	61.2433	**64.5629**	77.4537	**81.8942**
Mushroom	100.00	100.00	100.00	100.00	98.8042	**99.3589**	100.00	100.00
Nursery	98.2525	**98.9126**	99.1521	99.1521	91.5482	91.5482	93.6788	93.6788
Ticdata2000	83.3241	**86.3782**	82.4582	**86.4892**	78.9789	**82.3472**	84.5321	**87.7626**

The bold values are the classification accuracy with the best classification performance.

IEP in data sets Connect-4, Gene, Handwritten, Hepatitis, Letters, and Ticdata2000.

Accordingly, we can conclude that the incremental algorithm IEPDO can find a feasible attribute reduction.

Hence, the experimental results verified that the proposed incremental methods IEPAO and IEPDO can obtain an efficient attribute reduction and provide a quick data preprocessing method for dynamic data sets.

6. Conclusions and Further Study

Attribute reduction can effectively eliminate redundant information. Though the discernibility matrix method is one of the intuitive and effective reduction methods, it cannot deal with the reduction of large-scale data sets effectively because of memory overflow. The attribute reduction mechanism based on IEP can effectively solve the problem of space consumption analyzed in this paper. During the reduction process, IEP effectively prunes the subdivisions with cardinality 1, which speeds up the calculation of equivalence class division. IEP has better time and space effects in reduction, but it only adapts to the environment of static data sets. Considering the constant updating of data, in reality, IEPAO and IEPDO are proposed on the basis of IEP to deal with the reduction of adding data objects and deleting data objects respectively. As to IEP, the entire data set has to be reduced again and consumed a lot of time with the data changes. IEPAO and IEPDO only compute the changed part data and combine the previous reduction results, which can obtain the data set with fewer redundancies and better outcomes.

Of course, the algorithm proposed has some shortcomings in this paper. For example, (1) The IEP method can only reduce integer or character data, but cannot adapt to process other types of data. (2) The incremental update algorithm proposed in this paper does not consider the changes in attributes and values.

In the future, we will conduct the research from the following aspects: design an increment algorithm adapting to different types of data; develop a reduction method regarding the change values of data objects; propose an incremental mechanism with adding and deleting some attributes. Additionally, those approaches should adapt to an incomplete decision system.

Appendix

The proof of Theorem 5

Proof. From Definition 8, we have

$$
\begin{aligned}
\mathrm{IEP}_{U\cup U_{\Delta x}}(D|C) &= C^2_{|U\cup U_{\Delta x}|} - C^2_{|X_1'|} - C^2_{|X_2'|} - \cdots - C^2_{|Xk|} - C^2_{|X_{k+1}|} - C^2_{|X_{k+2}|} - \cdots - C^2_{|X_m|} - C^2_{|Y_{k+1}|} - C^2_{|Y_{k+2}|} - \cdots - C^2_{|Y_m'|} \\
&= \frac{\left[\left(|U| + |U_{\Delta x}|\right)\left(\left(|U| + |U_{\Delta x}| - 1\right) - |X_1'|\left(|X_1'| - 1\right) - |X_2'|\left(|X_2'| - 1\right) - \cdots - |X_k'|\left(|X_k'| - 1\right) - |X_{k+1}|\left|X_{k+1}\right| - 1\right) - |X_{k+2}|\left|X_{k+2}\right| - 1 - \cdots - |X_m|\left|X_m\right| - 1 - |Y_{k+1}|\left|Y_{k+1}\right| - 1 - |Y_{k+2}|\left|Y_{k+2}\right| - 1 - \cdots - \left|Y_{m_t}\right|\left(\left|Y_{m_t}\right| - 1\right)\right]}{2} \\
&= \frac{\left[\begin{array}{c} |U|(|U| - 1) + ||U_{\Delta x}|\left(|U_{\Delta x}| - 1\right) + 2|U||U_{\Delta x}| - |X_1|\left(|X_1| - 1\right) - |Y_1|\left(|Y_1| - 1\right) - 2|X_1||Y_1| - |X_2|\left(|X_2| - 1\right) - \\ |Y_2|\left(|Y_2| - 1\right) - 2|X_2||Y_2| - \cdots - |X_k|\left(|X_k| - 1\right) - |Y_k|\left(|Y_k| - 1\right) - 2|X_k||Y_k| - |X_{k+1}|\left(|X_{k+1}| - 1\right) - |X_{k+2}|\left(|X_{k+2}| - 1\right) - \cdots - \\ |X_m|\left(|X_m| - 1\right) - |Y_{k+1}|\left(|Y_{k+1}| - 1\right) - |Y_{k+2}|\left(|Y_{k+2}| - 1\right) - \cdots - \left|Y_{m_t}\right|\left(\left|Y_{m_t}\right| - 1\right) \end{array}\right]}{2} \\
&= \mathrm{IEP}_U(D|C) + \mathrm{IEP}_{U_{\Delta x}}(D|C) + |U||U_{\Delta x}| - \sum_{i=}^{k}|X_i||Y_i|.
\end{aligned}
\tag{A.1}
$$

□

Data Availability

All the data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the Natural Science Foundation of China (61836016), the Quality Improvement Project of Chaohu University on Discipline Construction (kj21gczx03), Special Support Plan for Innovation and Entrepreneurship Leaders in Anhui Province, the Provincial Natural Science Research Program of Higher Education Institutions of Anhui Province (KJ2021A1030), and the Key Subject Subprojects of Chaohu University ZDXK-201815.

References

[1] Z. Pawlak and A. Skowron, "Rudiments of rough sets," *Information Sciences*, vol. 177, no. 1, pp. 3–27, 2007.

[2] L. S. Riza, A. Janusz, C. Bergmeir et al., "Implementing algorithms of rough set theory and fuzzy rough set theory in the r package"roughsets," *Information Sciences*, vol. 287, pp. 68–89, 2014.

[3] V. S. Ananthanarayana, M. Narasimha Murty, and D. K. Subramanian, "Tree structure for efficient data mining using rough sets," *Pattern Recognition Letters*, vol. 24, no. 6, pp. 851–862, 2003.

[4] Y. Y. Yao and Y. Zhao, "Attribute reduction in decision-theoretic rough set models," *Information Sciences*, vol. 178, no. 17, pp. 3356–3373, 2008.

[5] Q. H. Hu, D. R. Yu, and Z. X. Xie, "Neighborhood classifiers," *Expert Systems with Applications*, vol. 34, no. 2, pp. 866–876, 2008.

[6] R. W. Swiniarski and A. Skowron, "Rough set methods in feature selection and recognition," *Pattern Recognition Letters*, vol. 24, no. 6, pp. 833–849, 2003.

[7] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. V. lahavas, "Multilable classification of music into emotions," in *Proceedings of the 9th International Conference on Music Information Retrieve(ISMIR2008)*, pp. 325–330, Philadelphia, PA, USA, 2008.

[8] G. Hao, L. Longshu, Y. Chuanjian, and D. Jian, "Incremental reduction algorithm with acceleration strategy based on conflict region," *Artificial Intelligence Review*, vol. 51, no. 4, pp. 507–536, 2019.

[9] L. Yin and Z. Jiang, "A fast attribute reduction algorithm based on a positive region sort ascending decision table," *Symmetry*, vol. 12, no. 7, p. 1189, 2020.

[10] Y. Jiang, "Minimal element selection in the discernibility matrix for attribute reduction," *Chinese Journal of Electronics*, vol. 28, no. 1, pp. 6–12, 2019.

[11] P. Sowkuntla and P. S. V. S. S. Prasad, "MapReduce based parallel fuzzy-rough attribute reduction using discernibility matrix," *Applied Intelligence*, vol. 52, no. 1, pp. 154–173.

[12] P. Wang, L. D. Qu, and Q. L. Zhang, "Information entropy based attribute reduction for incomplete heterogeneous data," *Journal of Intelligent and Fuzzy Systems*, vol. 43, no. 1, pp. 219–236, 2022.

[13] C. Wang, Y. Huang, W. Ding, and Z. Cao, "Attribute reduction with fuzzy rough self-information measures," *Information Sciences*, vol. 549, no. 12, pp. 68–86, 2021.

[14] C. Z. Wang, Y. Wang, M. W. Shao, Y. Qian, and D. Chen, "Fuzzy rough attribute reduction for categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 818–830, 2020.

[15] M. Akram, G. Ali, and J. C. R. Alcantud, "Attributes reduction algorithms for m-polar fuzzy relation decision systems," *International Journal of Approximate Reasoning*, vol. 140, no. 2022, pp. 232–254.

[16] M. Akram, G. Ali, and J. C. R. Alcantud, "Parameter reduction analysis under interval-valued m-polar fuzzy soft information," *Artificial Intelligence Review*, vol. 54, no. 7, pp. 5541–5582.

[17] Y. G. Jing, T. R. Li, J. F. Huang, and Y. Y. Zhang, "An incremental attribute reduction approach based on knowledge granularity under the attribute generalization," *International Journal of Approximate Reasoning*, vol. 76, pp. 80–95, 2016.

[18] X. H. Hu and N. Cercone, "Learning in relational database:A rough set approach," *Computational Intelligence*, vol. 11, no. 2, pp. 323–338, 1995.

[19] D. Ye and Z. J. Chen, "A new discernibility matrix and the computation of a core," *Acta Electronica Sinica*, vol. 30, no. 7, pp. 1086–1088, 2002.

[20] M. Yang and Z. H. Sun, "Improvement of discernibility matrix and the computation of a core," *Journal of Fudan University(Natural Science)*, vol. 43, no. 5, pp. 865–868, 2004.

[21] Z. Dong, M. Sun, and Y. Y. Yang, "Fast algorithms of attribute reduction for covering decision systems with minimal elements in discernibility matrix," *Int.J.Mach. Learn. & Cyber*, vol. 7, no. 2, pp. 297–310, 2016.

[22] W. Wei, J. Y. Liang, J. H. Wang, and Y. H. Qian, "Decision-relative discernibility matrices in the sense of entropies," *International Journal of General Systems*, vol. 42, no. 7, pp. 721–738, 2013.

[23] L. J. Li, M. Z. Li, J. S. Mi, and B. Xie, "A simple discernibility matrix for attribute reduction in formal concept analysis based on granular concepts," *Journal of Intelligent and Fuzzy Systems*, vol. 37, no. 3, pp. 4325–4337, 2019.

[24] M. Yang, "An incremental updating algorithm for attribute reduction based on improved discernibility matrix," *Chinese J.Compt*, vol. 30, no. 5, pp. 815–822, 2007, in Chinese.

[25] H. Ge, L. S. Li, and C. J. Yang, "Incremental attribute reduction based on simplified discernibility matrix," *J.Sichuan University(Eng.Sci.Edi)*, vol. 45, no. 1, pp. 116–124, 2013, in Chinese.

[26] Y. Liu, L. D. Zheng, Y. L. Xiu et al., "Discernibility matrix based incremental feature selection on fused decision tables,"

International Journal of Approximate Reasoning, vol. 118, pp. 1–26, 2020.
[27] W. Wei, P. Song, J. Y. Liang, and X. Y. Wu, "Accelerating incremental attribute reduction algorithm by compacting a decision table," *Int.J.Mach. Learn. & Cyber*, vol. 10, no. 9, pp. 2355–2373, 2019.
[28] J. B. Zhang, T. R. Li, D. Ruan, D. Liu, and C. B. Zhao, "Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems," *International Journal of Approximate Reasoning*, vol. 53, no. 4, pp. 620–635, 2012.
[29] F. M. Ma, M. W. Ding, T. F. Zhang, and J. Cao, "Compressed binary discernibility matrix based incremental attribute reduction algorithm for group dynamic data," *Neurocomputing*, vol. 344, pp. 20–27, 2019.
[30] S. Y. Zhuang and D. G. Chen, "A novel algorithm for the vertex cover problem based on minimal elements of discernibility matrix," *Int.J.Mach.Learn &Cyber.,* vol. 10, no. 12, pp. 3467–3474, 2019.
[31] P. Ni, S. Y. Zhao, X. Z. Wang, H. Chen, C. Li, and E. C. Tsang, "Incremental feature selection based on fuzzy rough sets," *Information Sciences*, vol. 536, pp. 185–204, 2020.