Applied Soft Computing Journal 203 (2026) 116041

Contents lists available at ScienceDirect

Applied Soft Computing

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

An efficient graph-driven information fusion approach using dominance rectangular-granular rough set for multi-source ordered decision information systems

Zishuo Yang [a], Guangming Xue [b], Weihua Xu [a,*]

[a] College of Artificial Intelligence, Southwest University, Chongqing, 400715, PR China
[b] College of Information Engineering, Nanning University, Nanning, 530200, PR China

HIGHLIGHTS

- We propose a novel information fusion method for multi-source ordered decision systems.
- The method uses a dominance-rectangular neighborhood rough set to evaluate attributes.
- Attributes from different sources are graphed with dependencies as path weights.
- Combining rough sets with an adjacency matrix significantly improves computational efficiency.

ARTICLE INFO

Keywords:
Graph-driven information fusion
Dominance-rectangular neighborhood
Rough set
Multi-source information systems

ABSTRACT

With the increasing size and complexity of data, information fusion has emerged as an effective approach to retain valid information in multi-source decision information systems. This study proposes a novel information fusion method for multi-source ordered decision information systems. The method significantly reduces the computational cost of rough sets through the use of rectangular rough sets. Compared with existing information fusion algorithms which focus on the traditional neighborhood rough set, this paper addresses the gap in how to represent distance relationships between samples without requiring large computational resources for distance calculation. The novelty of this method, mainly includes the following: i) This algorithm implicitly stores the distance relationships between samples by constructing a binary tree instead of using the traditional neighborhood rough set which requires calculating the distance between a sample and all other samples, thereby consuming substantial computational resources. ii) By using graph theory, the uncertainty of attributes can be stored, and this algorithm can represent the superiority relationship between attributes through the convergence of the infinite series of the graph matrix. Finally, the effectiveness and efficiency of the proposed method are validated through comparisons with seven other methods, including three common methods (Min, Mean, Max), on 12 datasets. The results demonstrate that the proposed method achieves substantial improvements in both effectiveness and efficiency.

1. Introduction

With the development of information technology in today's society, more and more novel information technologies have been developed, such as big data, cloud computing, 5G communication and so on. However, the scale of data processed in social life is increasing, and the sources of data are becoming more and more abundant, which undoubtedly brings great challenges to data analysis and data mining. Uncertainty metrics are a common tool for data analysis and knowledge discovery, which are often used to obtain valid data in information systems by analyzing data with uncertainty [1,2].

With the size and scale of data increasing, multi-source information systems are becoming an important data form that contains more information than traditional single source information tables. There are various applications of multi-source information systems: the urban flooding risk assessment using multi-source data [3], the diagnosis of

* Corresponding author.
Email addresses: yzs20040416@email.swu.edu.cn (Z. Yang), xueguangming0417@126.com (G. Xue), chxuwh@swu.edu.cn (W. Xu).

https://doi.org/10.1016/j.asoc.2026.116041
Received 11 March 2025; Received in revised form 2 June 2026; Accepted 12 July 2026
Available online 24 July 2026

the same patient in different hospitals [4], the different risk assessments of the same credit card from banks [5] or energy management [6]. In many cases, we are often faced with multi-source information systems, and this form of data is becoming more and more common in society, so conducting the data mining on this type of data is very necessary.

Rough set as a common method of data analysis and uncertainty measurement was first proposed by the Polish scientist Pawlak [7], through the calculation of the upper and lower approximations in order to achieve the purpose of quantifying the uncertainty of the data. Since its introduction, Rough set theory has been widely applied [8–10], and it has been extensively researched and applied in the areas of attribute reduction [11,12], classification problems [13,14], and uncertainty measurement [15,16] among others.

In order to better cope with the scale challenges brought about by multi-source decision information systems in the current field of data science and considering the unique advantages of rough sets in current data processing and knowledge discovery, the information fusion algorithms for multi-source information systems have become a research point attracting more attention, and the application of rough sets as an uncertainty measure in the environment of information fusion has been widely developed [17,18]. Sang et al. [19] proposed a multi-scale information fusion method inspired by a fuzzy rough combination entropy, Cai et al. [20] proposed another new information fusion model based on fuzzy approximate conditional entropy for interval-valued data, Yuan et al. [21] constructed an information fusion framework via K-nearest-based neighbor rough set for multi-source ordered decision systems. Pan [22] proposed a dynamic fusion approach via information entropy based on rough set. From the above studies, it can be seen that rough set theory has been widely studied and applied to different data backgrounds in the field of information fusion.

However, when reviewing the current field of information fusion in multi-source information systems, there are mainly two major issues that need to be addressed. Firstly, one of them is the huge computation required for traditional neighborhood granularity. Actually, the core of the rough set-based modeling framework often relies on the computation of granularity, and since Pawlak proposed his classical rough set, many scholars have developed rough set models based on more flexible granularity computation. Kong et al. [23] built a simplified rough set model by adding some new rules to the traditional rough set, Wu and his teammates [24] presented a neighborhood rough set framework based on a new notion of neighborhood equivalence relation for feature selection. Additionally, Wang et al. [25] proposed a feature selection model based on the weighted fuzzy rough set. Xu et al. [26] even constructed a rough set model with the combination of FCM and k-Nearest Neighbor. Li et al. [27] proposed a granular-ball-based rough set model for high-dimensional variation. The methods mentioned above inevitably require distance calculation between samples to determine the granularity of the rough set, which results in a huge amount of computational cost.

And the second one is that the existing information fusion algorithms are rarely aimed at multi-source ordered decision information systems. Xu et al. [28] take into account the interval length and the number of data points in the interval to construct a fusion method and Chen et al. [29] propose a novel information fusion and feature selection method based on information entropy for time-series data. There is an information fusion method combining improved entropy for multi-source interval-value decision information systems in the study of Xu [30] and Zhang et al. [31] proposed the dynamic fusion method with the increase of samples or attributes. From the above-mentioned related research, it can be found that most information fusion algorithms are oriented towards multi-source information systems with classical data, time series data or interval values. In fact, rough set theory has been extensively studied for ordered decision information systems, but these studies focus more on the construction of rough set models [32,33] or feature selection [10,34–36] for ordered decision information systems.

By reviewing the information fusion algorithms based on rough sets, it is found that there are still the following research gaps in the field of information fusion that need to be studied:

(i): At present, almost all information fusion algorithms based on rough sets rely heavily on the calculation of spatial distance between samples when calculating rough granularity, which often leads to an $O(n^2)$ computational cost that is often unacceptable for large data sets.
(ii): There are few research results on the information fusion of multi-source ordered decision information systems, but it has a wide range of applications in society.
(iii): Some algorithm frameworks based on rough sets are often constructed using information entropy [15,30,34], which frequently requires repeated calculations when implementing the algorithm.

In order to fill the research gap mentioned above, we introduce a new rough set calculation method: rectangular neighborhood rough set [37, 38], and adapt it to a rectangular dominance neighborhood rough set so that the information fusion algorithm proposed in this paper can meet the needs of multi-source ordered decision information systems.

However, since the rectangular neighborhood rough set only uses an approximate representation of the distance between samples instead of the true distance, we introduce graph theory to aggregate the distance information between samples in the algorithm framework to make the sample distance representation more accurate. Graph-based methods provide us with a new perspective for the information system. In the study of Giorgio Roffo et al. [39], the attributes can be seen as nodes, and the information between attributes can be stored in the edges of each node. Not only that, many graph-based methods have been developed [40] for feature selection or other applications. Nie et al. [41] put forward an unsupervised feature selection method by structured graph optimization, Zhang et al. [38] take the graph as the feature selection framework for interval-valued data, and Akhiat Y et al. [42] propose a novel feature selection method by the combination of graph theory and decision tree. So it can be seen that graph theory has been studied widely in data analysis. However, via the construction of graphs for information systems, the calculation efficiency can improve significantly. However, having much dependency on prior knowledge [43] may be a problem.

Roughly speaking, on the whole this work proposes an efficient information fusion approach for MS-ODIS, which is based on the combination of dominance-rectangular neighborhood rough set and the construction of a graph. Generally speaking, the method proposed by this paper takes the fusion process by selecting the best one of the attributes under different sources. More details about the implementation of this algorithm will be explained in the later part of this paper.

According to the previous research gaps summary of the existing information fusion algorithms, the primary contributions of this work are summarized as follows:

- The dominance-rectangular neighborhood rough set based on rectangular granularity for ordered decision information systems is proposed. Compared with the previous neighborhood rough set calculation method, this method can reduce the time complexity of distance calculation from $O(n^2)$ to $O(nlogn)$.
- We utilize the lower approximation based on the dominance-rectangular neighborhood as an indicator for attributes in multi-source ordered decision information systems (MS-ODIS). Furthermore, we employ the adjacency matrix of a graph to aggregate the significance of attributes across different sources.
- The proposed information fusion method is compared with several existing algorithms, including three common methods (Min, Mean, Max) and other rough set-based methods. The results demonstrate significant improvements in both effectiveness and runtime efficiency.

To make the progression of the subsequent content in the paper clearer, some basic concepts and theories used in the information fusion algorithm will be described in Section 2, the more details regarding the specific process and implementation of information fusion will be clearly illustrated in Section 3, relevant experiments comparing the proposed method with other algorithms in terms of effectiveness and runtime are conducted in Section 4, and the main content of the paper is summarized in the final Section 5.

2. Preliminaries

In this section, some of the concepts and theory that may be used in the information fusion framework proposed in this paper will be explained in detail to facilitate the understanding of the subsequent content.

2.1. Multi-source ordered decision information system

Firstly, the Decision Information System (DIS) should be classified. Let $DIS = (U, AT, V, f_{AT}, D)$ be a decision information system, where $U = \{x_1, x_2, \dots\dots, x_n\}$ is the sample set, and $AT = \{a_1, a_2, \dots\dots, a_m\}$is the attribute set, $f : U \times AT \Rightarrow V$ is the mapping function for the DIS, and D is the set of labels for the samples.

However, if the samples in the DIS could be sorted by the value of $f(x, a_m)$, where $a_m \in AT$ i.e., $f(x_i, a_m) \preceq f(x_j, a_m) \Longleftrightarrow x_i \preceq x_j$, then the attribute is denoted as a criterion. If $\forall a_m \in AT, a_m$ is a criterion, then the ordered decision information system (ODIS) could be denoted as $ODIS = \{U, AT, V, f_{AT}, D\}$.

After the above introduction for ODIS, a multi-source ordered decision information system (MS-ODIS) could be expressed as $MS-ODIS = \{(ODIS_i|ODIS_i = (U^i, AT^i, V^i, f^i_{AT}, D), i = 1, 2, \dots\dots, p\}$, where the MS-ODIS is a finite non-empty set in which every element is an ODIS.

The above is a description and elaboration of the informational background of the study in this paper: ODIS and MS-ODIS.

2.2. Introduction of approximation precision (AP) and approximation quality (AQ)

In this subsection, two critical indicators for a DIS, the approximation precision and approximation quality, will be classified. The two concepts were first studied by Pawlak in 2002 [44]. Their specific calculations for the two indicators are shown below.

A $DIS = (U, AT, V, f_{AT}, D)$ is given, where $U = \{x_1, x_2, \dots\dots, x_n\}$, $AT = \{a_1, a_2, \dots\dots, a_m\}$. Then the relation of different samples under the attribute subset $B \subseteq AT$ could be obtained as:

$$R_B = \{(x, y) : U \times U\}. \tag{1}$$

Please pay attention to the fact that the relation R_B is not deterministic, and the relations that are said to be used in this paper will be described in detail later. After making the definition of the relation R_b intelligible, $\forall X \in U$ the lower approximation and upper approximation under the relation R_B can be expressed as:

$$\underline{R_B}(X) = \{x \in U|[x]_{R_B} \subseteq U\}, \tag{2}$$

$$\overline{R_B}(X) = \{x \in U|[x]_{R_B} \cap U \neq \emptyset\}, \tag{3}$$

in which $[x]_{R_B}$ can be denoted as $[x]_{R_B} = \{y|(x, y) \in R_B\}$.

While reviewing the relevant information [28,30,45], we found that most authors use a similarity based on sample distance to induce sample relations as well as equivalence classes. Given that, we have the definition of the distance under the attribute subset B $dis_B(x_i, x_j)$ as follows:

$$dis_B(x, y) = \frac{\sum_{\forall a \in B} (f(a, x) - f(a, y))^2}{|B|}, \tag{4}$$

where $|B|$ is the number of attributes in B, $x, y \in U$.

The distance of different samples after normalization can be expressed as

$$dis^{nor}_B(x, y) = \frac{dis_B(x, y)}{max(dis(x, x_k))}, k = 1, 2, \dots\dots, n. \tag{5}$$

Given that the main research object of this paper is MS-ODIS, so when inducing the relation, we need to consider not only other samples that are similar to sample x_i, but also other samples that are superior to x_i. Thus, the dominance-neighborhood relation is

$$DNR_B = \{(x, y) : U \times U|dis^{nor}_B(x, y) < \alpha \wedge f(a, y) > f(a, x), \forall a \in B\}, \tag{6}$$

where α is the distance threshold, which is a parameter that needs to be set.

The upper and lower approximations induced from the dominance-neighborhood relation are respectively:

$$\underline{DNR_B}(X) = \{x \in U|[x]_{DNR_B} \subseteq X\}, \tag{7}$$

$$\overline{DNR_B}(X) = \{x \in U|[x]_{DNR_B} \cap X \neq \emptyset\}, \tag{8}$$

where the $[x]_{DNR_B}$ could be denoted as $[x]_{DNR_B} = \{y|xDNR_By\}$. Let $U/D = \{Y_1, Y_2, \dots\dots, Y_d\}$, then for a $ODIS = (U, AT, V, f_{AT}, D)$ the AP and AQ based on the $\underline{DNR}(X)$ under the attribute subset B can be denoted as:

$$AP_{DNR_B}(U/D) = \frac{|\sum_{i=1}^{d} \underline{DNR_B}(Y_i)|}{|\sum_{i=1}^{d} \overline{DNR_B}(Y_i)|}, \tag{9}$$

$$AQ_{DNR_B}(U/D) = \frac{|\sum_{i=1}^{d} \underline{DNR_B}(Y_i)|}{|U|}. \tag{10}$$

According to the findings of Pawlak [44], the AQ could represent the measurement of the approximation classification precision and AP could reflect the classification quality. The approximation precision and quality will improve with the AP and AQ increasing.

2.3. Granular-rectangular rough set

Here, the main methods of granular-calculation used in this paper will be explained in detail. Since Pawlak proposed rough set theory [7], other rough set methods have appeared, among which neighborhood rough set (NRS) has been widely studied [46], however, in the typical neighborhood rough set calculation, when calculating the neighborhood of one sample, it is always necessary to calculate the distance to all other samples, which causes computational inefficiency especially when dealing with large datasets.

However, according to the study of related work [38], the application of granular-rectangular rough set (GRRS) can greatly improve the calculation efficiency. Here we provide the definition of rectangular granularity.

Definition 1. $DIS = (U, AT, f_{AT}, D)$ is given as a decision information system, $\forall B \subseteq AT$ and $B = \{a1, a2, .., a_l\}, 1 \leqslant l \leqslant m$. G is used as the spatial partition of the attribute set B over the target samples, $G = \{G_1, G_2, \dots\dots, G_n\}$. The subspace $G_r (0 \leq r \leq n)$ for the samples universe U can be represented as $G_r = \{x_i|a_{1_{i-}} \leqslant f(a_1, x_i) \leq a_{1_{i+}}, a_{2_{i-}} \leqslant f(a_2, x_i) \leq a_{2_{i+}}, \dots\dots, a_{l_{i-}} \leqslant f(a_l, x_i) \leq a_{l_{i+}}\}$, where $a_{l_{i-}}$ and $a_{l_{i+}}$ are the upper and lower bounds of that space.

According to the description of rectangular granularity in Definition 1, each partition space G_r in G can be partitioned into finer granularity spaces in the same way as G, and the upper and lower bounds of each granularity rectangle will become tighter and tighter

as the space is refined. A hierarchical tree structure is formed during the continuous refinement of the rectangular granularity, in which all the data points in the subspace share the parent space to serve as the neighborhood, which then avoids the calculation of the distance to other samples. Unlike NRS which uses Euclidean distances, GRRS uses a special neighborhood radius, specifically, the neighborhood radius is a discrete value indicating the number of levels from the leaf node upward to its parent space, with each level of the neighborhood radius r increasing by one.

2.4. The graph theory

Similar to the article for the interval-valued decision information system [39] description and operation, ODIS can be described by constructing a complete undirected graph $G =< V, E >$, in which the node set $V = AT = \{a_1, a_2, \ldots\ldots, a_m\}$ represents the attribute set, and the edge set E represents the relationship between different features. The adjacency matrix N can be obtained by constructing the weight function $w(i, j)$ on the edges between feature a_i and a_j, in other words, the adjacency matrix N which is employed by graph G can be used as a method to sustain the intricate relationships between different nodes (features) and the information conveyed by the nodes in the information system, i.e., $N(i, j) = w(i, j)$.

Each element of the adjacency matrix N is determined by a weight function $w(.,.)$. According to the related work [38], there are two metrics that are adopted to help capture the relationship information between the feature a_i and a_j. More details could be expressed as follows:

$$w(i, j) = \delta \times I_{ij} + (1 - \delta) \times C_{ij}, \tag{11}$$

where I_{ij} can be understood as a metric reflecting the significance of the features to the whole information system, which demonstrates the contribution to the classification problem. The other one can be seen as a representation of the lack of correlation between features. The δ decides the weight of the two metrics.

However, attention here, in the subsequent study of this paper, we used the weight function $w(i, j)$ to describe the relationship of the same attribute between different sources, so we discarded the metric cor, and the final weight is expressed as

$$w(i, j) = I_{ij}, \tag{12}$$

and more details about the function $w(i, j)$ will be illustrated in section three.

2.5. Matrix infinite series

In the Section 2.4, the adjacency matrix is introduced. In the subsequent content of the paper, the computation for the matrix infinite series will be discussed.

Given $M_1, M_2, M_3, \ldots\ldots, M_k, \ldots\ldots$ as a sequence of matrices, where $M_k = m_{ij}^{(k)\in C^{m\times n}}$. The sum of the whole form: $M_1 + M_2 + M_3 + \ldots\ldots + M_k + \ldots\ldots$ can be named as a matrix power series (MPS) [47]. $\sum_{k=1}^{\infty} M_k$ can represent the whole form.

Let $(\mathbb{M})^P = \sum_{k=1}^{P} M_k$, in which P could be any integer and the whole formula represents the partial sum of the matrix sequence $\sum_{k=1}^{\infty} M_k$. If $lim_{P\to\infty} \sum_{k=1}^{P} M_k$ converges and has the limit $S = lim_{P\to\infty} \sum_{k=1}^{P} M_k$, it can be concluded that the matrix infinite series converges and $S = \sum_{k=1}^{\infty} M_k$.

Meanwhile, if the limit of the $\sum_{k=1}^{\infty} M_k$ is another matrix S, then each element in the matrix $M^{(P)}$ converges and has the limit $lim_{P\to\infty} m_{ij}^{(P)} = S_{ij}$. The details for the proof of the above properties can be found in the literature [38] and will not be repeated here.

The above mentioned properties of MPS greatly reduce the computational complexity and simplify the computation process when summing the matrix infinity series.

3. Information fusion method for the MS-ODIS

In this section of the paper, the information fusion method for MS-ODIS will be proposed, in which the general process can be demonstrated in Fig. 1, and the specific implementation of the process will be explained later.

3.1. Dominance-rectangular granularity based on the tree

By the Definition 1, the granular-rectangular can be obtained via the construction of a tree for the universe U. Here the exact process will be described.

An $ODIS = (U, AT, f_{AT}, D)$ is given, for the attribute a_m and $a_m \in AT$. It can be copied following the Algorithm 1.

So, the construction process for the tree has been summarized as the Algorithm 1, the time complexity and space complexity will be analyzed. Firstly, the sorting of sample attribute values consumes the majority of the time during the tree construction process. Generally, the time complexity for sorting is $O(nlogn)$ (by the quick sort method). After that, data accessing is necessary for one node's branching, which will consume time. When constructing the leaf nodes, data access needs to be performed n times. When level ups, the number of data accesses will become $n/2$ and so on. So the time complexity for the splitting of nodes into child nodes is $O(n + n/2 + n/4 + \cdots + 1) = O(2n - 1) = O(n)$. Thus, the time complexity for the whole process is $O(nlogn) + O(n) = O(nlogn)$.

Next, the space complexity for Algorithm 1 is as follows. Considering the leaf node of the tree, the number of nodes is n, but the number of contents in every node is 1. Similarly, there are $n/2$ nodes and 2 elements in the content for every node one level up and so on. It is concluded obviously that the space occupied by each layer's nodes is $O(n)$, and the number of levels is $O(logn)$. Therefore, the space complexity for Algorithm 1 is $O(nlogn)$.

Definition 2. An $ODIS = (U, AT, f_{AT}, D)$, and $a_m \in AT$, the tree based on the attribute a_m could be induced. For sample x_n in the leaf node, the node in its upward r levels can be called the rectangular granularity of sample x_n, which can be denoted as $P_r^m(x_n)$, where r is the radius of its special neighborhood and m represents the m-th attribute in the AT.

However, for rectangular granularity, the domain radius r does not precisely limit the domain range of the samples. It merely roughly indicates the size of the domain range while calculating the rectangular-neighborhood rough set. When r is smaller, the number of median splits increases, and the distance between the samples in the derived rectangular-neighborhood rough set becomes closer.

More details about the process of the instruction for the tree and Definition 2 can be acquired in the Fig. 2.

Definition 3. Let $ODSI = (U, AT, f_{AT}, D)$, $P_r^m(x_n)$ be the rectangular-granularity of sample x_n under the attribute a_m, where $a_m \in AT$. So, based on the rectangular-granularity, the relationship of dominance-rectangular neighborhood for the attribute $a \in AT$ can be denoted as

$$DRNR_a^r = \{(x, y) \in U \times U | y \in P_r(x) \wedge f(y, a_m) \geq f(x, a_m)\}, \tag{13}$$

Based on which, the equivalence classes can be obtained as follows

$$[x]_{DRNR_a} = \{y|(x, y) \in DRNR_a\}. \tag{14}$$

Definition 4. Let $ODIS = (U, AT, f_{AT}, D)$, the decision attribute can divide the sample set U into r equivalence classes: $U/D = \{D_1, D_2, \ldots\ldots, D_r\}$. According to the relationship in the Definition 3, the lower and upper approximations for one equivalence class under the attribute $a \in AT$ can be expressed as follows:

$$\underline{DRNR_a}(D_i) = \{x|[x]_{DRNR_a} \subseteq D_i\}, \tag{15}$$

$$\overline{DRNR_a}(D_i) = \{x|[x]_{DRNR_a} \cap D_i \neq \emptyset\}, \tag{16}$$

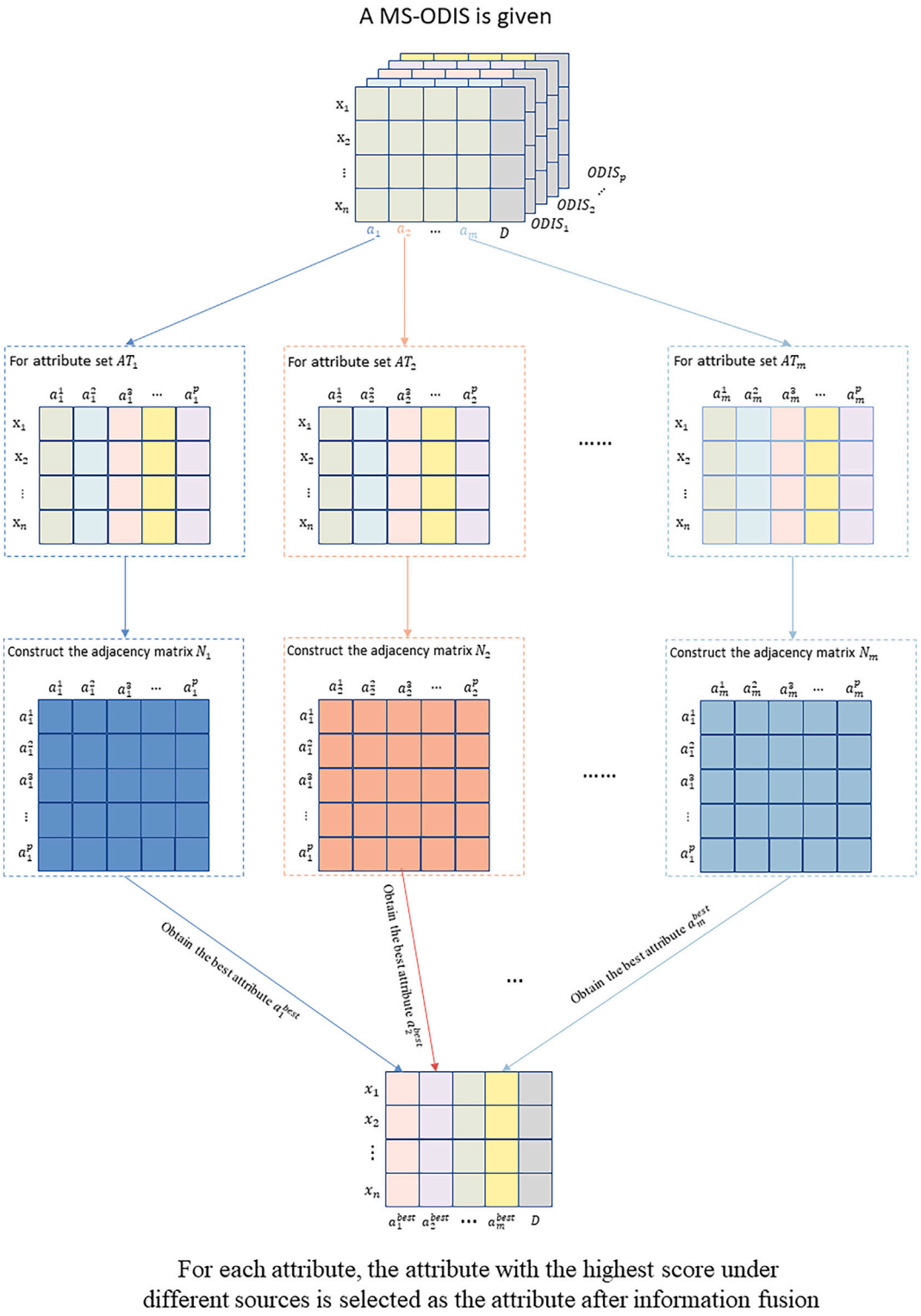

Fig. 1. The process of the information method for MS-ODIS.

And for the decision attribute D, the lower approximation and upper approximation can be obtained as follows:

$$\underline{DRNR_a}(D) = \bigcup_{i=1}^{r} \underline{DRNR_a}(D_i), \tag{17}$$

$$\overline{DRNR_a}(D) = \bigcup_{i=1}^{r} \overline{DRNR_a}(D_i). \tag{18}$$

And according to the rough set theory proposed by Pawlak [7] in 1998, the positive region is equivalent to the lower approximation, which contains samples that can be classified into one concept without any disagreement at all, based on some specific features. The positive region under the attribute a can be expressed as follows:

$$Pos_{DRNR_a}(D) = \underline{DRNR_a}(D) \tag{19}$$

Algorithm 1: BTC (Binary Tree Construction).

Input: A root node arr with **sorted** $arr.content = \{f(x_1, a_m), f(x_2, a_m), \ldots\ldots, f(x_n, a_m)\}$

Here, each node is represented by a tuple $v = (v.content, v.left, v.right, v.parent)$, where the $v.left$ and $v.right$ are pointed to the child node respectively, $v.content$ stores the attribute value $f(x, a_m)$ as a list, $v.parent$ is pointed to the parent node.

Output: The set of leaf nodes $Leaves$

```
if |arr.content| = 1 then
    return {arr};
end
mid ← ⌊|arr.content|/2⌋;
left_node ← new node;
right_node ← new node;
left_node.content ← arr.content[1 : mid];
right_node.content ← arr.content[mid + 1 : |arr.content|];
arr.left ← left_node;
arr.right ← right_node;
left_leaves ← BTC(left_node);
right_leaves ← BTC(right_node);
return left_leaves ∪ right_leaves;
```

Definition 5. Let $ODIS = \{U, AT, f_{AT}, D\}$, and the importance of the attribute a_i for the whole attribute set AT is denoted as:

$$Imp(a_i) = \frac{|Pos_{DRNR_a}(D)|}{|U|} \tag{20}$$

Here, it is necessary to clarify that I_{ij} in Eq. (12) is determined based on Eq. (20), and it can be expressed as

$$I_{ij} = \max(Imp(a_i), Imp(a_j)), \tag{21}$$

where $Imp(a_i)$ and $Imp(a_j)$ are normalized.

In this subsection, the calculation process of the dominance rectangular-neighborhood rough set has been detailedly presented. What truly distinguishes the method of this paper from other existing rough set fusion algorithms is the difference in the granularity calculation method. The latter, when calculating the granularity, often needs to compute the distance between each sample and all other samples, which often leads to $O(n^2)$ time complexity, while the former only requires $O(nlogn)$.

3.2. The fusion process based on the graph

In this section, the method of information fusion for MS-ODIS will be described. The information fusion method proposed in this paper is mainly based on the adjacency matrix between attributes, but in order to illustrate the construction of the adjacency matrix, it is first necessary to explain how the elements in the adjacency matrix are obtained.

For a weighted undirected fully connected graph $G =< V, E >$, the adjacency matrix N can be obtained from the graph. The element of N is $N_{ij} = I_{ij}$. It is important to note here that when constructing the adjacency matrix for MS-ODIS, it is constructed for the same attribute of different information sources, and then the same attribute under different sources is ranked based on the adjacency matrix. The ranking process within graph theory is achieved through the utilization of matrix power series properties, which comprehensively considers all path contributions between features in different sources, making the ranking more robust and adaptable.

Definition 6. Let $MS - ODIS = \{ODIS_i | ODIS_i = (U, AT^i, f^i_{AT}, D), i = 1, 2, \ldots\ldots, p\}$, where $AT^i = \{a^i_1, a^i_2, \ldots\ldots, a^i_m\}$, the a^p_m means the m-th attribute in the p-th source. To better explain the subsequent content, there is a definition that the same attributes $\{a^1_i, a^2_i, \ldots, a^p_i\}$ under different sources can be denoted as $AT_i = \{a^1_i, a^2_i, \ldots\ldots, a^p_i\}$. Then for the attribute set AT_i under the MS-ODIS, the element of the adjacency matrix N_i can be expressed as $N_i(k, h) = I^i_{kh} = \max(Imp(a^k_i), Imp(a^h_i))$, where k and $h \in \{1, 2, \ldots\ldots, p\}$.

Following this, the process of ranking the features between different sources based on the MPS will be illustrated clearly.

Definition 7. Let $G_i =< V, E >$ be a weighted undirected fully connected graph derived from the a_i attributes under different sources based on MS-ODIS, where nodes V represent the attribute set $\{a^1_i, a^2_i, \ldots\ldots, a^p_i\}$. Let $\delta_i = \{\vec{v_0} = a^k_i, \vec{v_1}, \ldots\ldots, \vec{v_{t-1}}, \vec{v_t} = a^h_i\}$ be a path of length t between the node $\vec{v_0}$ and $\vec{v_t}$. Then the weight of the path δ for the attribute a_i can be expressed as

$$w^{\delta_i}_{ij} = \prod_{a=0}^{t-1} N_i(\vec{v_a}, \vec{v_{a+1}}). \tag{22}$$

Generalizing this to the general case at this point, we denote Ω_i as the set of all paths of length t between node $\vec{v_0} = a^k_i$ and node $\vec{v_t} = a^h_i$ in

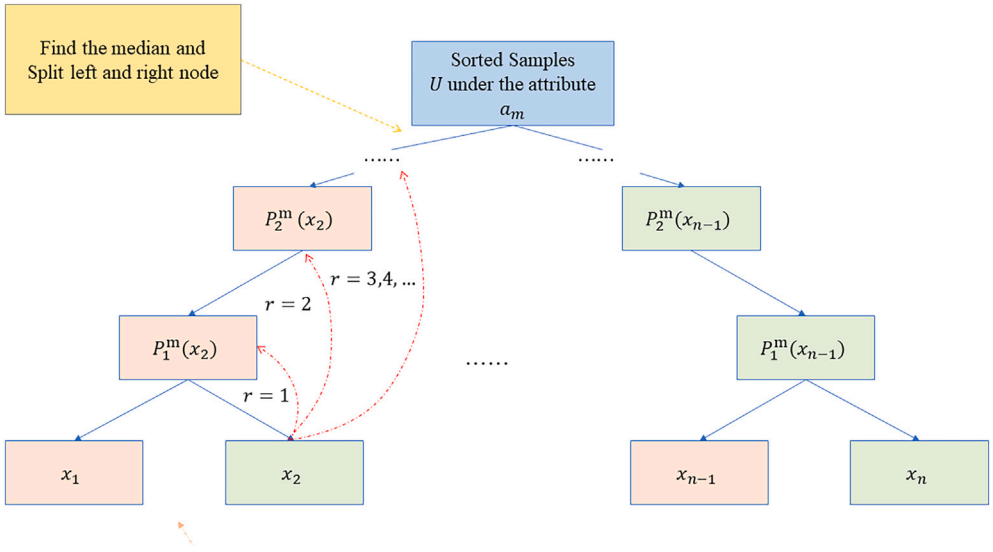

Fig. 2. The process of the rectangular-Granularity.

Table 1
An example of MS-ODIS is given.

U	$ODIS_1$			$ODIS_2$			$ODIS_3$			$ODIS_4$			d
	a_1^1	a_2^1	a_3^1	a_1^2	a_2^2	a_3^2	a_1^3	a_2^3	a_3^3	a_1^4	a_2^4	a_3^4	
x_1	0.73	0.15	0.58	0.34	0.92	0.21	0.87	0.05	0.43	0.12	0.68	0.79	1
x_2	0.29	0.64	0.93	0.51	0.07	0.84	0.38	0.72	0.15	0.66	0.23	0.55	1
x_3	0.88	0.47	0.11	0.62	0.03	0.95	0.19	0.56	0.27	0.74	0.31	0.08	0
x_4	0.05	0.82	0.67	0.23	0.45	0.76	0.94	0.17	0.39	0.49	0.88	0.53	1
x_5	0.36	0.19	0.85	0.77	0.54	0.28	0.63	0.91	0.04	0.97	0.42	0.61	0
x_6	0.52	0.27	0.69	0.18	0.86	0.37	0.81	0.48	0.93	0.09	0.75	0.24	0
x_7	0.94	0.08	0.25	0.67	0.33	0.59	0.44	0.72	0.16	0.83	0.50	0.91	1
x_8	0.60	0.71	0.03	0.89	0.14	0.47	0.22	0.57	0.80	0.35	0.68	0.12	0

the graph $G_i =< V, E >$, then the following sum equation:

$$\beta_i(k,h) = \sum_{\delta \in \Omega_i} w_{kh}^{\delta_i}, \tag{23}$$

In order to generalize the contribution of all the paths to the attribute a_i in different sources more comprehensively, the weights of the paths of length t are stored through a matrixN_i^t, in which the element can be written as follows:

$$N_i^t(k,h) = \beta_i(k,h). \tag{24}$$

It is noted that the feature a_i^k in the $ODIS_k$ and the feature a_i in the $ODIS_h$ source may be related directly or have other potential connections between the a_i attributes of other sources. Therefore, the score of the a_i attribute under source $ODIS_k$ for a fixed path length of t can be defined as follows:

$$\phi_i^t(k) = \sum_{h \in V} N_i^t(k,h), \tag{25}$$

Here, considering $t = 1$, for the attribute a_i, the minimum value of $N_i^1(k,h)_{k \in V}$ is $Imp(a_i)$ because of Eq. (21). So when calculating the $\phi_i^1(k)$, intuitively, if $Imp(a_i)$ is larger, the $\phi_i^1(k)$ is also larger. However, when $t = 1$, $\phi_i^1(k)$ only represents the score under path length $t = 1$. According to related work [38], to better capture the global influence of attribute a_i, where the path length t is from 1 to ∞, the score of attribute a_i over the path length $t > 1$: $\phi_i^t(k)$ should be computed. In other words, by calculating the adjacency matrix of different path lengths, not only are the aggregated attribute scores in the direct relationship considered, but also the potential relationships between different attributes are taken into account.

Also, for all path lengths, the score of attribute a_i under source $ODIS_k$ is defined as

$$\phi_i(k) = \sum_{t=1}^{\infty} \phi_i^t(k), \tag{26}$$

However it is not feasible to compute each term in the Eq. (26) one by one, here it can be written in the form of a matrix power series as follows

$$\phi_i(k) = \left[\left(\sum_{t=1}^{\infty} N_i^t \right) e \right]_k = \Psi_i e, \tag{27}$$

where the $\Psi_i = \sum_{t=1}^{\infty} N_i^t$ and e is a one-dimensional unit vector. However there is a problem that should be dealt is $\sum_{t=1}^{\infty} N_i^t$ may not converge. For this issue, the generating function [48,49] is adopted to add regularization which are typically used to assign a consistent value to a potentially divergent series. Then the Eq. (27) can be changed as:

$$\tilde{\phi}_i(k) = \left[\left(\sum_{t=1}^{\infty} \rho_i^t N_i^t \right) e \right]_k = \left[\left(\sum_{t=1}^{\infty} (\rho_i N_i)^t \right) e \right]_k = [\tilde{\Psi}_i e]_i, \tag{28}$$

where ρ_i is the regularization factor for N_i based on the attribute a_i in the different sources. The parameter ρ_i is denoted as $\rho_i = 0.9/radius(N_i)$, where $radius(N_i)$ is the spectral radius of matrix N_i.

It is important to explain why the value of ρ_i needs to be set to $0.9/radius(N_i)$. There is a Neumann Series Theorem in the theory of convergence of power series of matrices as $\sum_{t=1}^{\infty} N^t \Leftrightarrow radius(N) < 1 \Leftrightarrow \sum_{t=1}^{\infty} N^t = (E-N)^{-1} - E$, where E is the identity matrix. In the Eq. (28), $\tilde{\Psi}_i = \sum_{t=1}^{\infty} (\rho_i N_i)^t = (E - (\rho N_i)^{-1}) - E$ because $0 < radius(\rho_i N_i) = \rho_i \times radius(N_i) = 0.9/radius(N_i) \times radius(N_i) < 1$.

By the property of the MPS, the final score of the attribute a_i in the source $ODIS_k$ compared to attribute a_i under other sources can be denoted as

$$\tilde{\phi}_i(k) = [\tilde{\Psi}_i e]_k, \tag{29}$$

and for AT_i the best a_i^{best} attribute under all sources$\{ODIS_1, ODIS_2, \ldots\ldots, ODIS_p\}$ is $a_i^{best} = \underset{a_i^k \in AT_i}{argmax} \tilde{\phi}_i(k)$, then the attribute a_i^{best} in the MS-ODIS will be selected to form the final fused data table.

For the other attributes in MS-ODIS, the same procedure is used to select the best attribute from different sources to form the final fused data table.

In order to illustrate the fusion process, an example is given below to help illustrate the calculation of the definitions and the process.

Example 1. As seen in the Table 1, the$MS - ODIS = \{ODIS_1 = (U, AT^1, f_{AT}^1, D), ODIS_2 = (U, AT^2, f_{AT}^2, D), ODIS_3 = (U, AT^3, f_{AT}^3, D), ODIS_4 = (U, AT^4, f_{AT}^4, D)\}$, where $U = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$, $U/D = \{\{x_1, x_2, x_4, x_7\}\{x_3, x_5, x_6, x_8\}\}$. For the illustration of the subsequent calculations, it is specified here that $AT_m = \{a_m^1, a_m^2, \ldots\ldots, a_m^p\}$. For example, in this example $AT_1 = \{a_1^1, a_1^2, a_1^3, a_1^4\}$. For the purpose of the subsequent description, we present the data in Tables 1 as 2. Here we will choose the attribute set a_1^1 for the construction of the tree to illustrate in turn the acquisition of the rectangular-granularity and the calculation of the dominance-rectangular neighborhood, which can be seen in the Fig. 3. In the example and subsequent experiment the radius r is set as $r = 1$.

For the AT_1:

$$Pos_{DRNE_{a_1^1}} = \{x_1, x_2, x_4, x_5, x_6, x_7\}, Pos_{DRNE_{a_1^2}} = \{x_1, x_2, x_4, x_5, x_7, x_8\}$$

$$Pos_{DRNE_{a_1^3}} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}, Pos_{DRNE_{a_1^4}} = \{x_1, x_3, x_4, x_5\}$$

The $Imp(*)$ under the AT_1 after normalization:

$$Imp(a_1^1) = 0.5, Imp(a_1^2) = 0.5$$

$$Imp(a_1^3) = 1, Imp(a_1^4) = 0$$

For the AT_2:

$$Pos_{DRNE_{a_2^1}} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}, Pos_{DRNE_{a_2^2}} = \{x_1, x_2, x_5, x_7\}$$

$$Pos_{DRNE_{a_2^3}} = \{x_1, x_2, x_3, x_4, x_5, x_6\}, Pos_{DRNE_{a_2^4}} = \{x_3, x_4, x_7, x_8\}$$

The $Imp(*)$ under the AT_2 after normalization:

$$Imp(a_1^1) = 1, Imp(a_1^2) = 0$$

Table 2
The MS-ODIS after changed is given.

U	AT_1				AT_2				AT_3				d
	a_1^1	a_1^2	a_1^3	a_1^4	a_2^1	a_2^2	a_2^3	a_2^4	a_3^1	a_3^2	a_3^3	a_3^4	
x_1	0.73	0.34	0.87	0.12	0.15	0.92	0.05	0.68	0.58	0.21	0.43	0.79	1
x_2	0.29	0.51	0.38	0.66	0.71	0.07	0.72	0.23	0.93	0.84	0.15	0.55	1
x_3	0.88	0.62	0.19	0.74	0.47	0.03	0.56	0.31	0.11	0.95	0.27	0.08	0
x_4	0.05	0.23	0.94	0.49	0.82	0.45	0.17	0.88	0.67	0.76	0.39	0.61	1
x_5	0.36	0.77	0.63	0.97	0.19	0.54	0.91	0.42	0.85	0.28	0.04	0.53	0
x_6	0.52	0.18	0.81	0.09	0.27	0.86	0.48	0.75	0.69	0.37	0.93	0.24	0
x_7	0.94	0.67	0.44	0.83	0.08	0.33	0.72	0.50	0.25	0.59	0.16	0.91	1
x_8	0.60	0.89	0.22	0.35	0.64	0.14	0.57	0.68	0.03	0.47	0.80	0.12	0

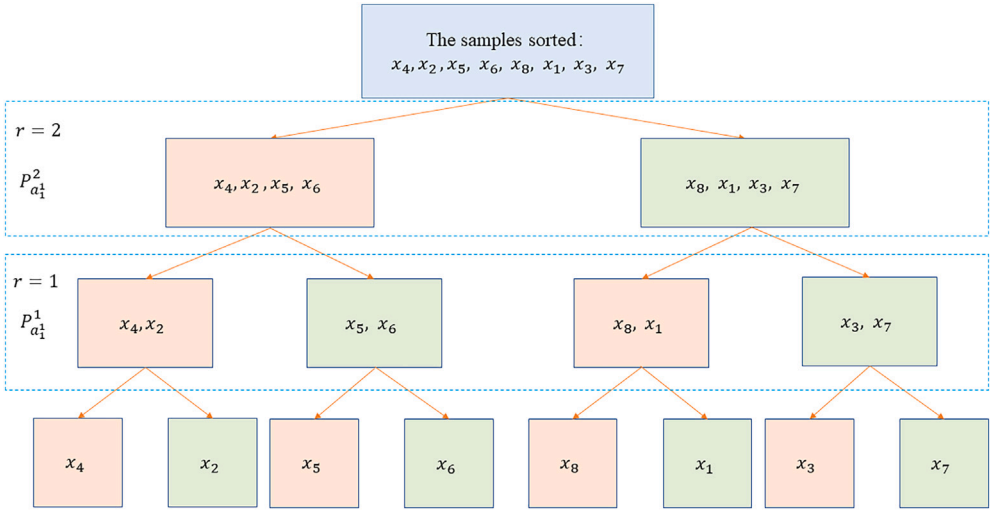

Fig. 3. The rectangular-granularity for attribute a_1^1.

$Imp(a_1^3) = 0.5, Imp(a_1^4) = 0$

For the AT_3:

$Pos_{DRNE_{a_3^1}} = \{x_1, x_2, x_3, x_6\}, Pos_{DRNE_{a_3^2}} = \{x_3, x_4, x_5, x_6, x_7, x_8\}$

$Pos_{DRNE_{a_3^3}} = \{x_1, x_2, x_3, x_4, x_6, x_8\}, Pos_{DRNE_{a_3^4}} = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$

The $Imp(*)$ under the AT_1 after normalization:

$Imp(a_1^1) = 0, Imp(a_1^2) = 0.5$

$Imp(a_1^3) = 0.5, Imp(a_1^4) = 1$

The three adjacency matrices for the attribute subset AT_1, AT_2, AT_3 are as follows:

$$N_1 = \begin{pmatrix} 0.5 & 0.5 & 1 & 0.5 \\ 0.5 & 0.5 & 1 & 0.5 \\ 1 & 1 & 1 & 1 \\ 0.5 & 0.5 & 1 & 0 \end{pmatrix} \quad N_2 = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0.5 & 0 \\ 1 & 0.5 & 0.5 & 0.5 \\ 1 & 0 & 0.5 & 0 \end{pmatrix}$$

$$N_3 = \begin{pmatrix} 0 & 0.5 & 0.5 & 1 \\ 0.5 & 0.5 & 0.5 & 1 \\ 0.5 & 0.5 & 0.5 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

The eigenvalues and spectral radius are calculated for the three adjacency matrices as follows:

$\lambda_1 = \{2.9161, 0.0000, -0.2622, -0.6538\}$

$\lambda_2 = \{2.6861, -1.0000, -0.1861, 0.0000\}$

$\lambda_3 = \{2.9161, -0.6538, -0.2622, 0.0000\}$

Then according to Eq. (28), the matrices $\tilde{\Psi}_1, \tilde{\Psi}_2, \tilde{\Psi}_3$ for the AT_1, AT_2, AT_3 can be calculated respectively as follows:

$$\tilde{\Psi}_1 = \begin{pmatrix} 1.794 & 1.794 & 2.742 & 1.554 \\ 1.794 & 1.794 & 2.742 & 1.554 \\ 2.742 & 2.742 & 3.954 & 2.375 \\ 1.554 & 1.554 & 2.375 & 1.213 \end{pmatrix} \quad \tilde{\Psi}_2 = \begin{pmatrix} 4.473 & 2.362 & 3.153 & 2.362 \\ 2.362 & 1.055 & 1.576 & 1.055 \\ 3.153 & 1.576 & 2.105 & 1.576 \\ 2.362 & 1.055 & 1.576 & 1.055 \end{pmatrix}$$

$$\tilde{\Psi}_3 = \begin{pmatrix} 1.213 & 1.554 & 1.554 & 2.375 \\ 1.554 & 1.794 & 1.794 & 2.742 \\ 1.554 & 1.794 & 1.794 & 2.742 \\ 2.375 & 2.742 & 2.742 & 3.955 \end{pmatrix}$$

Finally, the score of every source on the three attributes a_1, a_2, a_3 can be presented as follows by the Eq. (29):

$\tilde{\phi}_1 = [7.8857, 7.8857, 11.8159, 6.6978]$

$\tilde{\phi}_2 = [12.3521, 6.0506, 8.4129, 6.0506]$

$\tilde{\phi}_3 = [6.6978, 7.8857, 7.8857, 11.8159]$

Then the three attributes a_1, a_2, a_3 for all sources, the best one is as follows:

$$a_1^{best} = \underset{a_1^k \in AT_1}{argmax} \tilde{\phi}_i(k) = a_1^3$$

$$a_2^{best} = \underset{a_2^k \in AT_2}{argmax} \tilde{\phi}_i(k) = a_2^1$$

$$a_3^{best} = \underset{a_3^k \in AT_3}{argmax} \tilde{\phi}_i(k) = a_3^4$$

Algorithm 2: The information fusion algorithm(**FGIF**) for MS-ODIS.

Input: $MS-ODIS=\{ODIS_p|ODIS_i=(U,AT^i,f^i_{AT},D),i=1,2,\dots\dots,p\}$, $U=\{x_1,x_2,\dots\dots,x_n\}$, $AT^i=\{a^i_1,a^i_2,\dots\dots,a^i_m\}$, U/d $=\{d_1,d_2,\dots\dots,d_r\}$

Output: The fused data table $ODIS_{fused}=(U,AT_{fused},f_{fused},D)$

1 $AT_{fused}\leftarrow\{\}$.
2 **for** $j=1$ *to* m **do**
3 Construct a matrix $N_j(m\times m)$
4 **for** $i=1$ *to* p **do**
5 Obtain the dominance-rectangular neighborhood granularity of attribute a^i_j based on the construction of the tree.
6 Calculate $Imp(a^i_j)$ according to the Eq. 20.
7 **end**
8 **for** $k=1$ *to* p **do**
9 **for** $h=k$ *to* p **do**
10 $I_{kh}\leftarrow\max(Imp(a^k_j),Imp(a^h_j))$
11 $N_j(k,h)=I_{kh}$
12 $N_j(h,k)=I_{kh}$
13 **end**
14 **end**
15 $\rho_j=0.9/radius(N_j)$
16 $\tilde{\Psi}_j=(E-\rho_jN_j)^{-1}-E.$
17 $\tilde{\phi_j}=\tilde{\Psi}_je$
18 $a^{best}_j=\underset{a^k_j\in AT_j}{argmax}\tilde{\phi_j}(k)$
19 $AT_{fused}\leftarrow AT_{fused}\cup\{a^{best}_j\}$
20 **end**
21 return $ODIS_{fused}=(U,AT_{fused},f_{fused},D)$.

At last, the fused attribute set is $AT_{fused}=\{a^3_1,a^1_2,a^4_3\}$, and the fused data table is $ODIS_{fused}=(U,AT_{fused},f_{fused},D)$.

The above information fusion process for MS-ODIS is then summarized as an algorithm named **FGIF**, which is shown in Algorithm 2. Here we need to carry out a detailed analysis of the time complexity of Algorithm **FGIF**, through the content of Algorithm 2, which mainly produces time complexity from the operations in 4–7 lines, 8–14 lines, and line 18. Specifically, lines 4–7 primarily involve building the tree which includes operations of sorting and searching resulting the time complexity of $O(p\times nlogn)$. In lines in 8–14 operations mainly involve assigning values to the matrix, with a time complexity of $O(p^2)$, line 18 primarily deals with sorting the scores of the sources a_j under different attribute of sorting, and its time complexity depends on the algorithm used, generally it can be considered $O(plogp)$. Since the above process is applied to m attributes, the total time complexity of the entire process can be expressed as $O(m\times p\times nlogn)+O(p^3)+O(p^2logp)$. Nevertheless, in most cases, the sample quantity accounts for the majority, so the total time complexity simplifies to $O(m\times p\times nlogn)$. However, to achieve rapid retrieval of the rough sets of samples, a balanced binary tree needs to be maintained during the algorithm implementation. The space complexity of **FGIF** is $O(nlogn)$ as summarized in Algorithm 1.

In order to better demonstrate the superiority of the algorithm proposed in this study in terms of time complexity, the time complexity of some information fusion algorithms (which were also compared in the experimental part of this paper) using traditional neighborhood rough sets was compared with that of this algorithm.

The time complexity of Cai [20] is $O(n\times p\times m\times(r+n)+m\times p)$. The time complexity of Lin [28] is $O(p\times n^2)$. The time complexity of Pan [22] is $O(p\times m\times n^2+m)$. The time complexity of Chen [31] is $O(p\times m\times n\times(r+n)+m)$. From the above analysis of the algorithm's time complexity, it can be seen that the currently existing rough set-based feature selection algorithms have a time complexity of $O(n^2)$ in terms of n e.g., the number of samples. However, the algorithm proposed in this paper has a relationship with the sample size of $O(nlogn)$, which demonstrates the superiority of this algorithm in terms of time complexity.

3.3. Analysis of applicability and robustness

Applicability: The method proposed in this paper is an information fusion method for multi-source ordered decision information systems. In fact, as stated in the introduction of this paper, with the rapidly developing information society, the amount of data processed in daily life is becoming increasingly large. Data often appears in multiple sources in most cases. As one of the important types of multi-source ordered information systems, the research on its information fusion method is very necessary. For example, different educational institutions rate students based on their height, weight, grades, and award status for the same group of students; different financial institutions evaluate the investment risk and return of the same financial product; or different hospitals analyze the condition of the same patient. The information processed in these multi-source situations is often much larger. If we really want to understand the true situation of the target, we often need to process the information from all information sources. The information fusion algorithm proposed in this paper greatly reduces the complexity of data in multi-source situations and discovers more valuable information, which shows that the algorithm proposed in this paper has great applicability.

Robustness: Actually, this algorithm exhibits high robustness for noisy data, which is attributed to the advantageous calculation method of the rectangular neighborhood rough set presented in this paper. The granularity calculation method proposed in this paper for constructing binary trees through median partitioning is not an exact calculation of granularity. Therefore, when the data contains noise, such as when multiple sensors have discrepancies due to noise, or when human-collected data contains errors, etc., it will not affect the final calculation result of the algorithm. This also demonstrates that this paper exhibits good robustness for noise in the data.

4. Experimental analysis

In this subsection, aiming to demonstrate the effectiveness and efficiency of the proposed information fusion algorithm, 12 comparative experiments using 12 datasets are conducted. The 12 datasets are obtained from the UCI (University of California, Irvine Machine Learning Repository) database (https://archive.ics.uci.edu/datasets) and kaggle(https://www.kaggle.com/) database. Concrete information about these datasets can be found in Table 3.

The experimental setting, including the software and hardware environments, is presented in Table 4.

The object that the framework developed in this paper deals with is MS-ODIS, however, it is obvious that the datasets used in the experiments cannot be directly used as MS-ODIS, the following description explains how to process the data so that it can be applied to the proposed method:

(1): Firstly, the attributes without order relations should be removed, such as name, ID and so on. The numerical attributes are retained. For the categorical attributes, their order relations are determined by the semantics of the attributes and the values of the categorical attributes are changed to integer values—the higher the order, the higher the number.
(2): Max-min normalization is adopted for the datasets, and the values of the attributes are mapped to the range 0–1.
(3): For the single source of the datasets Table 3, constructing multiple sources is necessary. The detailed operation involves adding noise to 50% of the data in the datasets, as shown in Eq. (30).

Table 3
The summary of datasets.

No.	Datasets	Abbreviation	Samples	Attributes	Classes
Data1	Toxity	Tox	171	1203	2
Data2	Wine	Wine	178	13	3
Data3	Breast cancer wisconsin	BCW	569	30	2
Data4	Maternal health risk	MHR	1013	6	3
Data5	QSAR androgen receptor	QSAR	1687	1024	2
Data6	Wireless indoor localization	Wir	2000	7	4
Data7	Abalone	Aba	4176	8	3
Data8	Electrical grid stability simulated Data	Ele	10,000	14	2
Data9	Student placement	Stu	100,000	25	2
Data10	Diabetes health indicators	DHI	253,680	21	2
Data11	Indicators of heart disease 2020	Ind	319,795	17	2
Data12	Credit card fraud detection 2023	Cre	568,630	30	2

Table 4
The details of the software and hardware.

Name	model	Parameter
System	Windows 11	64bit
CPU	AMD Ryzen 5 5600H	3.3GHz
Platform	Python	python3.9
Program environment	Pycharm	Pycharm 2022
Memory	DDR4	16GB;3200MHz

$$F(x_i,a)=\begin{cases} f(x_i,a)+\eta, & \text{if } x_i \text{ in } U_{random} \\ f(x_i,a), & \text{else} \end{cases}, \tag{30}$$

where U_{random} is the samples set that need to be randomized to add noise η is the noise that obeys a mean of 0 with a variance of 0.1.

4.1. The analysis of parameter r

In this subsection, the effect of the parameter neighborhood radius r on the information fusion results is analyzed. According to Eq. (20), the importance of an attribute is assessed by the number of samples in the positive region, however different selections of r may yield different results. Therefore, the analysis of the parameter r is necessary.

In order to illustrate that different neighborhood radii r will have different effects on the results, here r is set to a step of 1 from 1 to 5, and at same time, the α is set to a step of 0.05 from 0.05 to 0.5. Observe how the results change when α is set to a step of 0.05 from 0.05 to 0.5 with different r. The result of AP is shown as Fig. 4 and the result of AQ is shown as Fig. 5.

The results in Figs. 4 and 5 are analyzed here, and on the whole, the trends of AP and AQ values with r and α changes are roughly the same. Both decrease as α decreases, where the change in r mainly affects the values of AQ and AP when α is smaller. Because, when r is relatively small, the dominance-rectangular granularity of all attributes derived at this time will be very fine, at this time according to Eq. (20) the importance of all attributes does not differ much, when it is relatively large, the dominance-rectangular granularity of all attributes derived will be very coarse again. Only when r is at a suitable value will part of the attribute's importance remain very high, while part of the attribute's importance will decrease.

4.2. The comparison of effectiveness for different algorithms

In Section 4.1, the paper analyzes the parameter r. A suitable r may lead to better information fusion. In the next subsection, the algorithmic framework studied in this paper is compared with three common information fusion methods: **Max**, **Mean**, **Min** and other information fusion algorithms:**Cai** [20], **Lin** [28], **Pan** [22], **Chen** [31] on AP, AQ and metric AE, respectively. Here, the weighted average results of the classification accuracy, recall rate and F1 value are obtained as the Average Effectiveness (AE) indicators, which can provide a more comprehensive presentation of the classification results of the classifier. The specific formula is as follows

$$AE=\frac{accuracy+recall+F1}{3}$$

However, it is noted that **Cai**, **Lin**, **Pan**, and **Chen** the four algorithms were not proposed for ordered information decision systems, so some changes are made when calculating the granularity dominance relationship. The basic introduction and parameters for the algorithms compared are presented here:

- **Min**: The minimum value of different sources in the same attribute is the final result after fusion, which can be denoted as $f_{fused}(a_i,x)=\min(f(a_i^k,x)), a_i^k\in AT_i$.
- **Mean**: The mean value of different sources in the same attribute is the final result after fusion, which can be denoted as $f_{fused}(a_i,x)=mean(f(a_i^k,x)), a_i^k\in AT_i$.
- **Max**: The maximum value of different sources in the same attribute is the final result after fusion, which can be denoted as $f_{fused}(a_i,x)=\max(f(a_i^k,x)), a_i^k\in AT_i$.
- **Cai**: The parameter δ is set as 0.25. In the method [30] proposed by Cai et al. the Eq. 9 in the paper [30] is changed as $if f(a,y)>f(a,x), Sim_a(x,y)=\frac{1}{1+dis_a(x,y)}, else, Sim_a(x,y)=0$.
- **Lin**: The parameter β is set as 1. The tolerance relation in the paper [28] is changed as $T_a=\{(x_i,x_j)|\frac{dis_a(x_i,x_j)}{max_{y\in U}dis(x_i,y)}\wedge f(a,x_j)>=f(a,x_i)\}$.
- **Pan**: The parameter k is set as 0.25.
- **Chen**: The parameter α is set as 1. The setting of the tolerance relation in the paper [31] is the same as that in **Lin**.

Comparison of AP and AQ: Under different values of r ranging from 1 to 5, the best results obtained by the FGIF algorithm proposed in this paper will be compared with other algorithms. The detailed comparison results for AQ and AP are presented in Figs. 6 and 7.

By comparing the AP and AQ of the FGIF algorithm with other algorithms under 12 datasets, it is clear that the FGIF algorithm has better outcomes in most datasets for both metrics. In a few datasets, although its performance may not be the best, it still has a high value of effectiveness. By analyzing Figs. 6 and 7, it can be concluded that, in terms of AP and AQ, the FGIF information fusion algorithm proposed in this paper matches or even outperforms other algorithms. Through comparison, it is apparent that the FGIF algorithm achieves better results in terms of both AP and AQ.

Comparison of AE value: The AE value of MS-ODIS after being fused under different algorithms is also compared. Five typical classifiers—K-nearest neighbor classifier (KNN), probabilistic neural network classifier (PNN), support vector machine (SVM), random forest (RF), and decision-making tree (DT)—are used in the fused data table. The parameter of KNN k is set as 5, the parameter σ of PNN is set as 0.1, the linear kernel is used by SVM, and the number of trees n in RF is set as 100. Five fold cross-validation is adopted in the classification. The results of the five classifiers for 12 datasets under different algorithms are shown in Table 5–9 respectively.

As shown in Tables 5–9, it can be seen that among a total of 60 cases under PNN, KNN, RF, SVM, DT, the FGIF algorithm has a better performance than other algorithms in 34 cases. Even in another 26 cases, its outcomes is still competitive, though not reaching the highest value. Therefore, taking into account the results of all the classifiers, the FGIF algorithm has the best performance compared to other methods.

Static analysis: To further illustrate the advantages of the FGIF algorithm proposed in this paper, the classification accuracy under the other

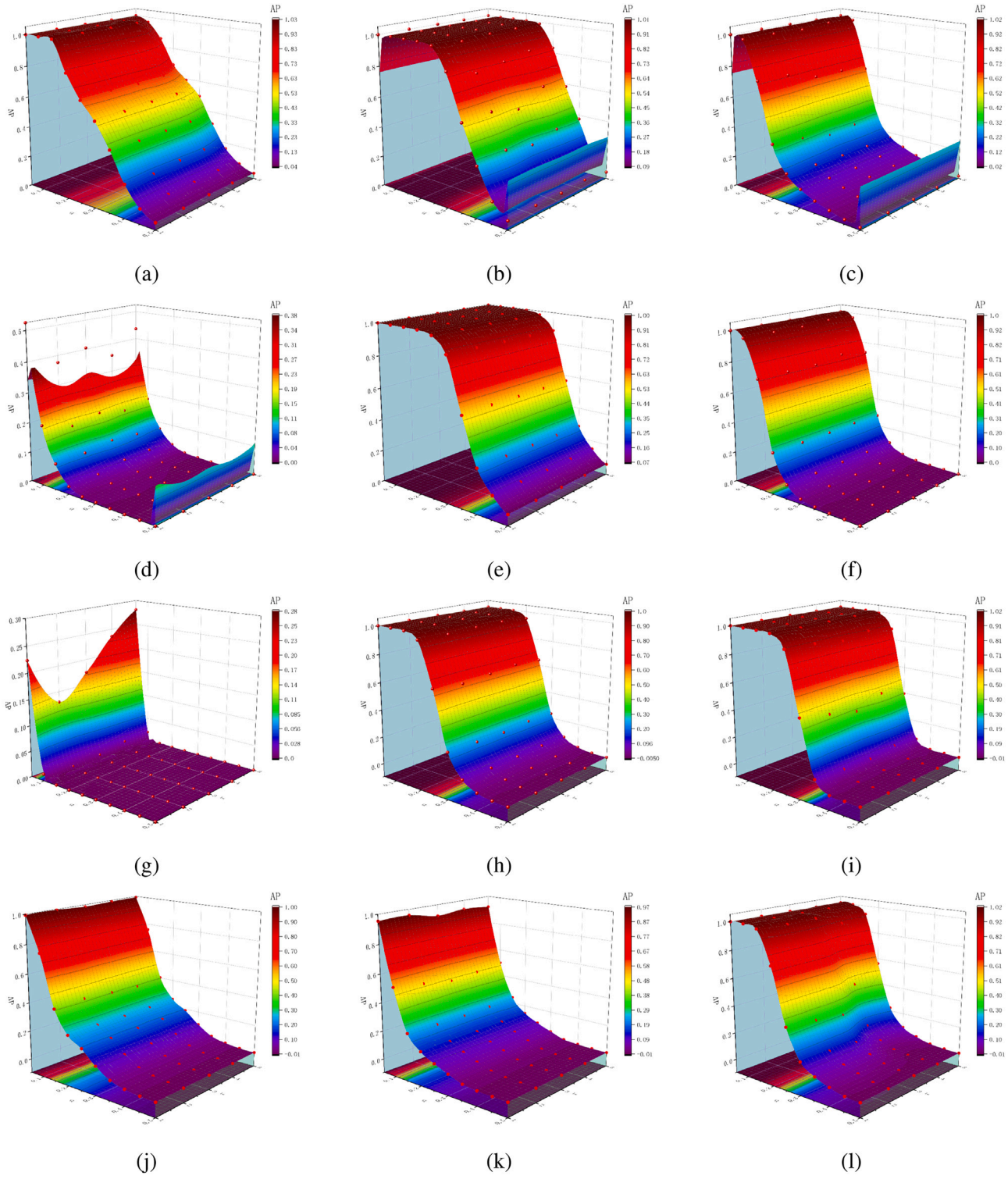

Fig. 4. The AP value of different radii r and α: (a)Tox, (b)Wine, (c)BCW, (d)MHR, (e)QSAR, (f)Wir, (g)Aba, (h)Ele, (i)Stu, (j)DHI, (k)Ind, (l)Cre.

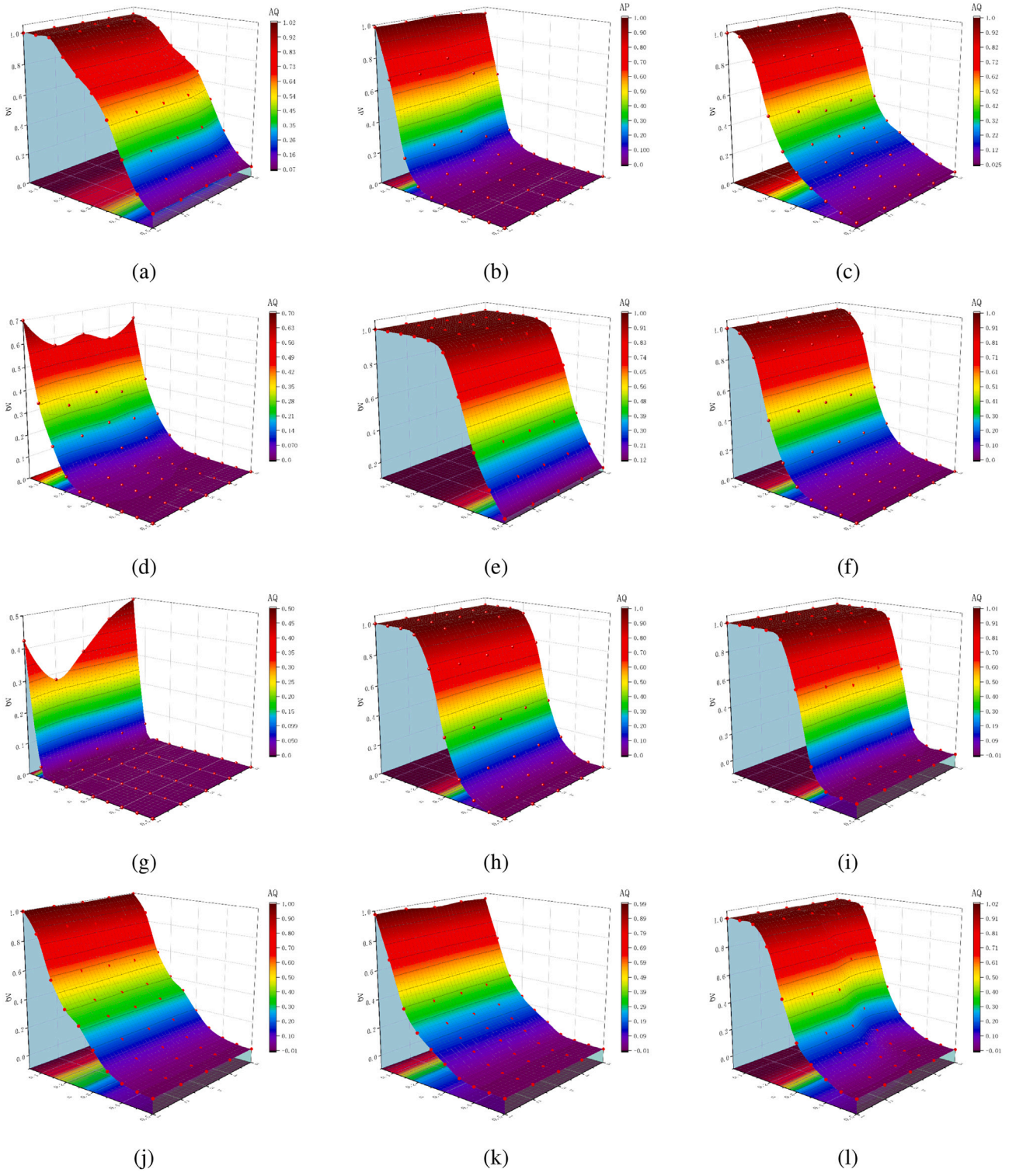

Fig. 5. The AQ value of different radii r and α: (a)Tox, (b)Wine, (c)BCW, (d)MHR, (e)QSAR, (f)Wir, (g)Aba, (h)Ele, (i)Stu, (j)DHI, (k)Ind, (l)Cre.

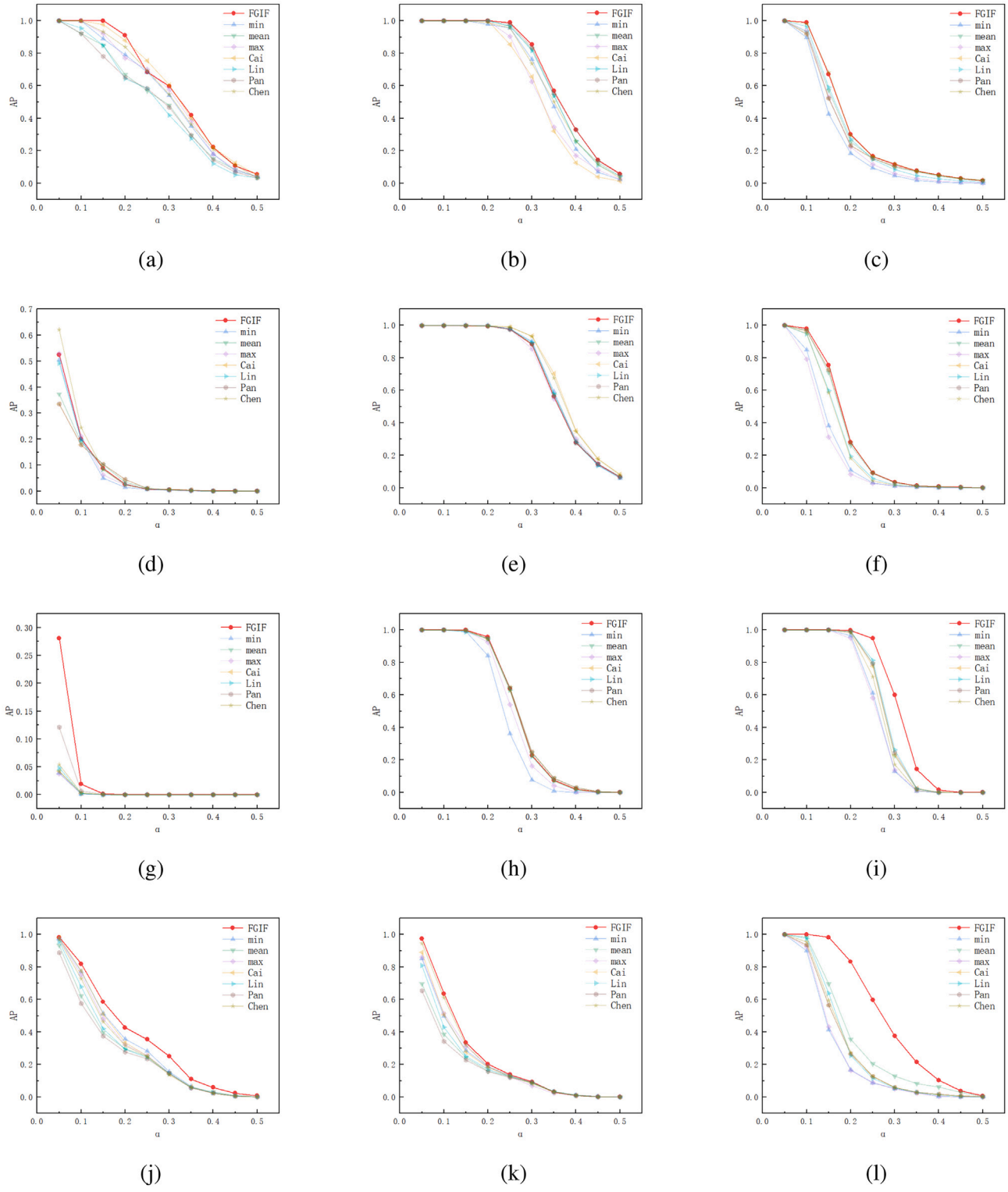

Fig. 6. The AP comparison with other algorithms: (a)Tox, (b)Wine, (c)BCW, (d)MHR, (e)QSAR, (f)Wir, (g)Aba, (h)Ele, (i)Stu, (j)DHI, (k)Ind, (l)Cre.

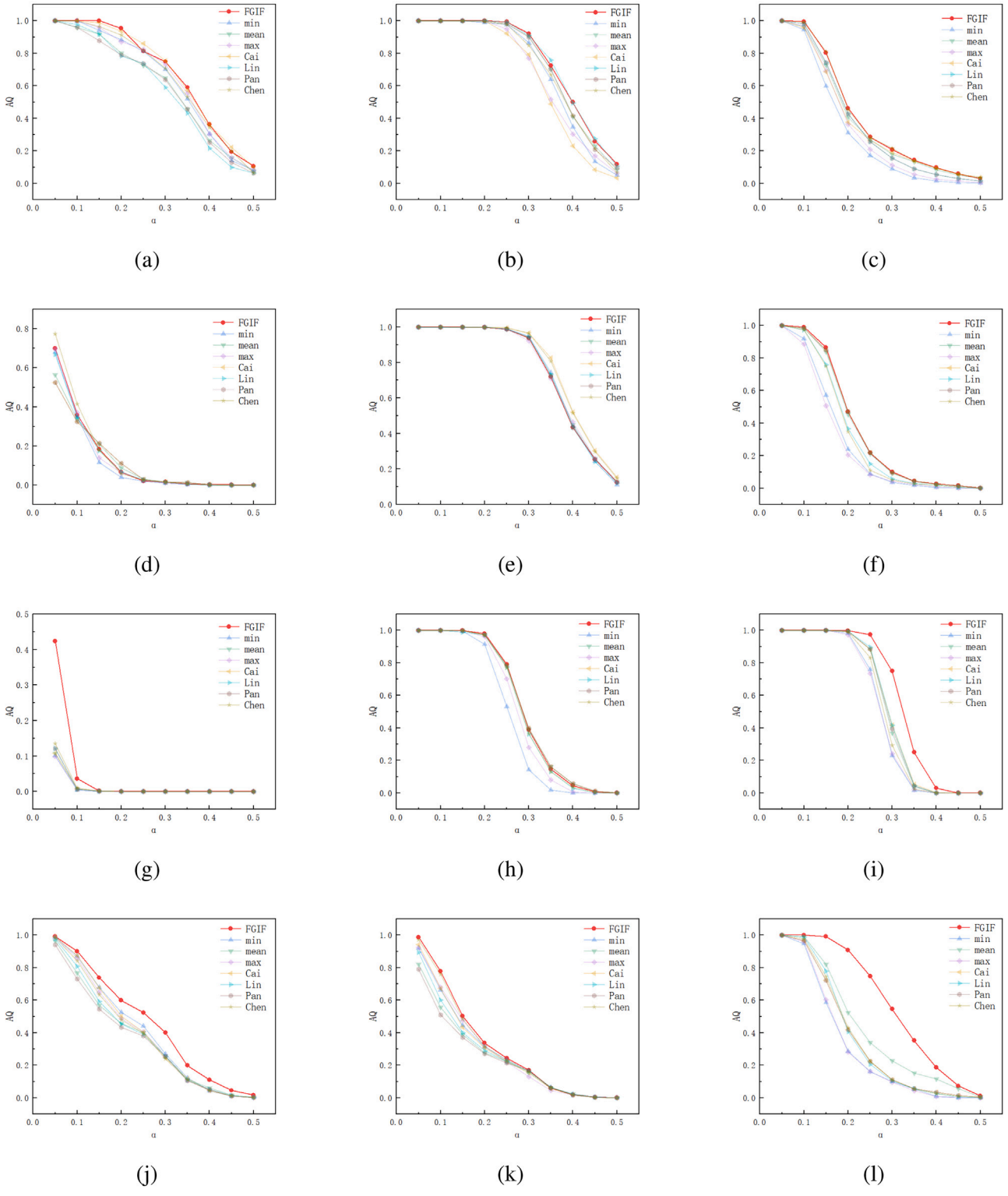

Fig. 7. The AQ comparison with other algorithms: (a)Tox, (b)Wine, (c)BCW, (d)MHR, (e)QSAR, (f)Wir, (g)Aba, (h)Ele, (i)Stu, (j)DHI, (k)Ind, (l)Cre.

Table 5
The mean classification accuracy and variance (%) of fused data on KNN.

Dataset	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIF
Data1	59.61 ± 4.56	63.73 ± 4.75	62.90 ± 3.97	62.69 ± 5.53	62.91 ± 5.38	64.25 ± 5.74	63.46 ± 7.60	**64.59 ± 4.50**
Data2	92.68 ± 4.11	96.63 ± 2.57	95.47 ± 2.20	97.76 ± 2.29	94.92 ± 2.07	97.19 ± 2.29	**98.31 ± 2.85**	97.03 ± 1.13
Data3	92.93 ± 3.23	**96.82 ± 0.50**	92.23 ± 1.83	96.65 ± 1.03	94.70 ± 2.53	96.65 ± 1.03	95.94 ± 0.91	96.82 ± 1.24
Data4	66.47 ± 4.18	68.87 ± 0.45	65.70 ± 2.10	65.11 ± 3.34	66.53 ± 1.91	**69.51 ± 3.34**	64.86 ± 5.10	66.85 ± 1.77
Data5	89.70 ± 2.51	89.53 ± 2.43	**90.13 ± 2.28**	90.06 ± 2.49	90.01 ± 2.69	89.87 ± 2.09	89.62 ± 2.41	90.13 ± 2.12
Data6	95.70 ± 0.85	98.15 ± 0.43	94.95 ± 0.90	98.30 ± 0.68	97.05 ± 0.46	97.65 ± 0.68	96.40 ± 0.34	**98.60 ± 0.68**
Data7	53.02 ± 0.75	52.05 ± 4.30	52.95 ± 1.23	51.03 ± 0.91	51.91 ± 1.63	52.96 ± 0.91	51.41 ± 1.42	**53.56 ± 1.57**
Data8	87.08 ± 0.75	90.84 ± 0.80	87.51 ± 0.53	91.25 ± 0.09	87.03 ± 0.49	**91.25 ± 0.09**	90.19 ± 0.35	90.19 ± 0.42
Data9	50.70 ± 1.64	49.22 ± 1.33	51.81 ± 1.11	50.73 ± 0.77	**52.85 ± 0.80**	50.93 ± 1.04	51.76 ± 1.21	51.75 ± 1.46
Data10	83.90 ± 0.78	83.95 ± 0.80	83.69 ± 1.03	84.21 ± 0.60	84.04 ± 0.81	83.62 ± 0.07	83.81 ± 0.61	**84.43 ± 0.97**
Data11	89.71 ± 0.62	88.70 ± 0.68	88.46 ± 0.70	88.48 ± 0.59	88.69 ± 0.64	90.28 ± 0.43	**95.72 ± 0.77**	92.24 ± 0.38
Data12	94.24 ± 0.29	95.16 ± 0.44	94.34 ± 0.44	95.70 ± 0.74	94.88 ± 0.73	95.96 ± 0.74	95.50 ± 0.60	**95.96 ± 0.15**

Table 6
The mean AE value and variance (%) of fused data on PNN.

Dataset	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIF
Data1	59.98 ± 4.18	56.98 ± 4.43	59.01 ± 3.95	61.24 ± 3.42	54.82 ± 5.38	59.94 ± 5.74	59.95 ± 7.60	**61.98 ± 3.42**
Data2	92.71 ± 1.37	95.05 ± 2.17	**96.58 ± 1.41**	95.50 ± 2.23	95.50 ± 2.07	95.59 ± 2.29	96.09 ± 2.85	96.03 ± 2.25
Data3	94.18 ± 1.69	95.95 ± 1.13	92.60 ± 1.20	95.77 ± 1.44	93.48 ± 2.53	96.12 ± 1.03	95.96 ± 0.91	**96.35 ± 1.19**
Data4	68.00 ± 3.07	**72.41 ± 1.36**	66.37 ± 2.40	66.76 ± 3.26	70.39 ± 1.91	72.01 ± 3.34	67.39 ± 5.10	72.14 ± 4.04
Data5	86.63 ± 0.23	86.63 ± 0.34	86.37 ± 0.23	86.37 ± 0.23	86.55 ± 2.69	86.63 ± 2.09	86.59 ± 2.41	**86.63 ± 0.12**
Data6	95.40 ± 0.94	97.60 ± 2.21	95.05 ± 1.19	97.50 ± 0.70	96.60 ± 0.46	97.83 ± 0.68	95.80 ± 0.34	**97.85 ± 0.77**
Data7	51.00 ± 1.14	52.18 ± 2.74	51.85 ± 1.53	51.81 ± 0.58	51.98 ± 1.63	52.55 ± 0.91	53.19 ± 1.42	**53.24 ± 0.61**
Data8	84.35 ± 0.52	87.27 ± 0.38	84.00 ± 0.68	88.22 ± 0.80	83.82 ± 0.49	88.22 ± 0.09	86.57 ± 0.35	**88.22 ± 0.55**
Data9	50.28 ± 0.75	49.00 ± 1.36	50.49 ± 2.12	50.08 ± 1.56	**51.77 ± 0.80**	51.20 ± 1.04	51.59 ± 1.21	51.23 ± 1.87
Data10	78.58 ± 0.40	77.74 ± 1.32	77.18 ± 1.68	77.59 ± 1.04	77.09 ± 0.81	77.83 ± 0.07	77.97 ± 0.61	**78.66 ± 0.79**
Data11	81.87 ± 0.39	79.46 ± 0.76	81.24 ± 0.98	80.43 ± 1.07	80.62 ± 0.64	**82.47 ± 0.43**	81.99 ± 0.77	82.44 ± 1.46
Data12	94.48 ± 0.53	94.68 ± 0.88	94.70 ± 0.81	**95.00 ± 0.88**	94.90 ± 0.73	94.84 ± 0.74	94.86 ± 0.60	94.88 ± 0.67

Table 7
The mean AE value and variance (%) of fused data on RF.

Dataset	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIF
Data1	58.23 ± 6.98	57.22 ± 10.04	59.32 ± 4.19	63.26 ± 1.33	57.00 ± 2.19	59.81 ± 2.95	59.87 ± 6.13	**65.07 ± 3.76**
Data2	95.53 ± 4.15	98.87 ± 2.08	98.30 ± 3.26	97.76 ± 2.76	94.34 ± 2.09	97.75 ± 1.38	97.21 ± 1.13	**98.87 ± 2.23**
Data3	93.65 ± 1.07	95.23 ± 1.50	92.77 ± 2.67	96.47 ± 1.42	94.70 ± 1.90	96.83 ± 1.42	97.00 ± 1.62	**97.30 ± 1.54**
Data4	69.04 ± 2.16	70.65 ± 4.18	65.71 ± 0.99	69.97 ± 2.41	70.31 ± 2.22	**85.82 ± 2.51**	69.93 ± 3.05	80.55 ± 3.26
Data5	88.59 ± 2.35	88.52 ± 2.15	89.00 ± 1.83	89.12 ± 2.13	88.57 ± 2.24	**90.14 ± 2.18**	89.05 ± 2.22	89.99 ± 2.10
Data6	96.25 ± 0.48	98.05 ± 0.37	94.76 ± 0.86	98.25 ± 0.43	96.60 ± 0.43	98.15 ± 0.25	97.40 ± 0.43	**98.25 ± 0.25**
Data7	53.81 ± 1.23	54.52 ± 1.13	53.82 ± 0.64	54.31 ± 1.11	54.39 ± 0.67	55.17 ± 1.19	54.32 ± 0.60	**55.58 ± 0.96**
Data8	94.89 ± 0.23	98.56 ± 0.27	94.81 ± 0.64	99.99 ± 0.02	94.85 ± 0.28	99.99 ± 0.02	100.00 ± 0.02	**100.00 ± 0.02**
Data9	53.76 ± 0.92	53.46 ± 1.08	**55.69 ± 2.05**	54.69 ± 1.87	54.43 ± 0.68	53.23 ± 0.81	53.60 ± 1.40	54.00 ± 2.09
Data10	84.96 ± 0.49	84.38 ± 1.02	84.26 ± 0.48	84.32 ± 0.78	84.37 ± 0.98	85.14 ± 0.05	84.11 ± 1.17	**85.29 ± 1.21**
Data11	90.22 ± 0.60	88.85 ± 1.91	89.03 ± 0.71	89.07 ± 0.57	89.44 ± 0.34	90.47 ± 0.51	**97.82 ± 0.45**	89.00 ± 0.29
Data12	95.02 ± 0.71	95.82 ± 0.32	95.32 ± 0.35	97.74 ± 0.56	95.94 ± 0.73	**97.86 ± 0.37**	97.70 ± 0.49	97.84 ± 0.05

Table 8
The mean AE value and variance (%) of fused data on SVM.

Dataset	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIF
Data1	47.78 ± 4.63	49.39 ± 7.14	47.91 ± 5.39	50.14 ± 10.03	48.10 ± 4.50	48.72 ± 6.52	49.96 ± 7.30	**55.54 ± 7.18**
Data2	95.76 ± 3.24	97.74 ± 2.23	95.62 ± 3.54	97.76 ± 1.37	96.06 ± 1.76	97.74 ± 1.14	**98.87 ± 1.81**	98.61 ± 1.13
Data3	95.41 ± 1.88	96.99 ± 1.52	93.63 ± 2.83	97.17 ± 1.24	94.69 ± 1.42	97.35 ± 1.43	97.35 ± 1.52	**97.82 ± 1.41**
Data4	60.92 ± 2.08	61.94 ± 2.32	59.43 ± 2.37	61.59 ± 2.73	60.59 ± 1.27	**63.36 ± 2.93**	62.68 ± 0.56	63.35 ± 2.37
Data5	86.70 ± 2.67	86.73 ± 2.25	86.72 ± 2.65	86.28 ± 2.78	86.82 ± 2.41	86.86 ± 2.38	86.67 ± 2.41	**86.86 ± 2.47**
Data6	95.89 ± 0.58	97.60 ± 0.40	95.55 ± 0.91	97.95 ± 0.56	96.75 ± 0.43	97.90 ± 0.41	96.80 ± 0.56	**97.95 ± 0.09**
Data7	51.18 ± 0.94	50.84 ± 1.09	50.66 ± 0.83	50.96 ± 0.57	50.90 ± 0.84	51.24 ± 0.78	51.12 ± 0.75	**51.28 ± 1.12**
Data8	99.13 ± 0.15	98.92 ± 0.18	98.86 ± 0.83	99.21 ± 0.07	98.85 ± 0.14	99.21 ± 0.07	99.17 ± 0.10	**99.27 ± 0.10**
Data9	54.32 ± 0.57	51.89 ± 0.92	53.32 ± 1.66	50.30 ± 1.13	53.15 ± 2.04	53.99 ± 1.46	50.74 ± 2.09	**54.92 ± 1.01**
Data10	83.66 ± 0.46	84.12 ± 0.92	83.25 ± 0.59	83.25 ± 0.55	83.25 ± 0.70	83.77 ± 0.19	83.25 ± 0.70	**84.25 ± 1.17**
Data11	**90.05 ± 0.57**	89.13 ± 1.47	89.00 ± 0.69	89.00 ± 0.36	89.00 ± 0.36	90.01 ± 0.36	90.02 ± 0.36	90.03 ± 0.39
Data12	94.82 ± 0.64	94.88 ± 0.47	95.16 ± 0.48	95.22 ± 0.74	95.18 ± 0.61	95.30 ± 0.74	**96.27 ± 0.73**	95.36 ± 0.11

Table 9
The mean AE value and variance (%) of fused data on DT.

Dataset	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIF
Data1	53.68 ± 3.95	56.32 ± 6.67	55.55 ± 7.08	58.17 ± 7.62	58.22 ± 6.80	52.87 ± 5.91	58.79 ± 1.39	**60.66 ± 5.52**
Data2	89.37 ± 4.46	90.47 ± 2.18	88.24 ± 2.30	90.47 ± 2.45	89.88 ± 3.74	91.21 ± 2.97	**92.72 ± 4.77**	91.41 ± 1.71
Data3	89.43 ± 2.40	91.72 ± 1.04	88.38 ± 2.47	93.31 ± 2.19	91.37 ± 3.25	93.44 ± 2.87	93.31 ± 2.11	**94.20 ± 2.89**
Data4	63.12 ± 2.06	68.75 ± 1.85	59.76 ± 2.31	62.51 ± 3.23	65.89 ± 3.06	76.72 ± 3.81	62.17 ± 1.89	**77.09 ± 2.43**
Data5	85.17 ± 1.89	83.01 ± 2.50	85.14 ± 2.59	83.47 ± 1.98	84.04 ± 2.78	85.22 ± 2.27	83.37 ± 1.35	**86.37 ± 2.23**
Data6	90.74 ± 1.28	96.40 ± 1.25	90.44 ± 0.77	96.50 ± 0.34	94.45 ± 0.76	96.50 ± 0.69	95.75 ± 0.37	**96.60 ± 0.96**
Data7	48.26 ± 1.98	48.37 ± 0.87	47.90 ± 1.38	48.38 ± 1.07	49.50 ± 0.89	**49.59 ± 1.73**	48.27 ± 1.18	49.27 ± 0.35
Data8	93.96 ± 0.43	97.83 ± 0.33	94.15 ± 1.38	99.98 ± 0.02	93.95 ± 0.16	99.98 ± 0.02	99.98 ± 0.02	**99.98 ± 0.02**
Data9	**52.29 ± 1.99**	50.89 ± 0.56	51.70 ± 0.80	50.92 ± 1.11	51.64 ± 1.04	51.24 ± 1.32	50.41 ± 1.62	51.64 ± 1.95
Data10	**80.14 ± 1.07**	79.13 ± 1.50	79.21 ± 0.71	79.17 ± 1.35	78.84 ± 1.25	79.88 ± 0.09	79.98 ± 0.98	80.06 ± 0.75
Data11	85.26 ± 0.64	85.00 ± 1.31	85.08 ± 1.21	85.14 ± 1.75	84.88 ± 0.66	**86.31 ± 1.05**	85.62 ± 0.88	85.86 ± 0.58
Data12	91.88 ± 0.62	93.72 ± 0.27	93.18 ± 0.53	95.58 ± 1.00	93.00 ± 0.92	95.54 ± 0.89	95.64 ± 0.72	**95.72 ± 0.08**

Table 10
The static analysis result of different algorithms.

classifiers	algorithms								χ_F^2	F_F	$p-$value
	Min	Mean	Max	Cai	Lin	Pan	Chen	FGIG			
KNN	6.17	4.33	5.75	4.25	5.08	3.25	4.67	2	16.2411	2.6366	2.30×10^{-2}
PNN	5.33	4.92	6.17	4.92	5.42	2.92	3.75	1.67	15.2896	2.4477	3.24×10^{-2}
RF	6.08	5.08	6	3.67	5.67	2.92	4.25	2	25.9590	4.9197	5.12×10^{-4}
SVM	5.25	4.83	6.67	4.42	5.83	2.75	3.75	1.33	20.5989	3.5739	4.41×10^{-3}
DT	5.33	5.42	6.25	4.17	5.5	2.92	4	1.58	19.0190	3.2195	8.12×10^{-3}

fusion algorithms is statistically studied here. In order to achieve this, the Friedman test is used to compare the classification performance of different fusion algorithms. The Friedman test [50] can be denoted as:

$$\chi_F^2 = \frac{12N}{k(k+1)}\left(\sum_{j=1}^{k} R_j^2 - \frac{k(k+1)^2}{4}\right),$$

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2},$$

where, N and k are the number of datasets and algorithms in the experiment, and R_j is the average ranking of the algorithms on all datasets by the classification. F_F represents the $F-$distribution with$(k-1)$ and $(k-1)(N-1)$ degrees of freedom. The critical difference can be described as:

$$CD_\alpha = q_\alpha \sqrt{\frac{k(k+1)}{6N}},$$

The acquirement [20] of the average ranking value is as follows: the best-ranking value under accuracy metrics is set as 1, the 2nd-ranking value is set as 2 and so on. According to the Tables 5–9 the ranking value of each algorithm under every dataset could be obtained. Then the average ranking value of each algorithm can be calculated. Finally the value of χ_F^2 and F_F can be computed.

The specific results of Friedman test for KNN, PNN, RF, SVM and DT are shown in the Table 10, in which FGIF algorithm have the best ranking value and P-value both lower than 0.05 under all algorithms. It has been known that number of algorithms $k = 8$, number of datasets $N = 12$, and the critical point $q_{0.05}$ in the Nemenyi test is 2.569, so critical difference $CD_{0.05} = 2.569$. For further illustrating the difference between the 8 fusion algorithms, the CD chart Fig. 8 is used to to connect methods that do not di?er signi?cantly from each other, and it is possible to clearly illustrate the extent of the differences between the different algorithms. As can be seen in the Fig. 8, FGIF has the best average ranking for all classifiers and is significantly better than the algorithms Min, Max, Lin, Chen. Generally speaking, the FGIF algorithm proposed in this paper outperforms other algorithms in terms of classification effectiveness under the Friedman test.

4.3. The comparison of efficiency for different algorithms

In addition to the effectiveness of an algorithm, the efficiency of its running is also an important criterion. In this subsection, the running efficiency of FGIF algorithm will be compared with the other four algorithms: Lin, Pan, Chen and Cai, and the running time of each algorithm on each dataset is shown in Table 11, and its visualization is shown in Fig. 9.It should be noted that due to the large sample sizes of Data12 and Data10, during the experiments conducted by the Cai, Chen, Pan algorithm, the sampling method was adopted to extract the samples. Then, based on the relationship between the algorithm's time complexity and the samples, a proportion factor was multiplied to finally obtain the algorithm's running time.

According to the Fig. 9 and Table 11, it can be obvious to obtain the efficiency of FGIF algorithm. The FGIF algorithm has a lower running time than the other algorithms on 11 datasets.Even though the efficiency is not the highest on Dataset9, it still has strong competitiveness compared to the other algorithms. Compared to Lin, the FGIF algorithm is slightly more efficient than Lin, and compared to Pan, Chen, Cai, the FGIF algorithm runs much more efficiently than these algorithms, especially in large samples datasets. In contrast, the FGIF algorithm calculating the dominance-rectangular granularity through the construction of the tree, compared with other algorithms with calculating rough set of other types, its time complexity is much lower than other algorithms. If more data samples are used in the implementation, the efficiency advantage of the algorithm proposed in this paper will be greater. However, this advantage can only be achieved by storing a binary tree of $O(nlogn)$ size. Considering the current level of memory devices, this trade-off of time is worthwhile.

Here the effectiveness and efficiency of the algorithms are comprehensively reviewed, **Cai** and **Pan** are closer to the FGIF algorithm proposed in this paper in terms of fusion effectiveness, but their running efficiency is much lower than that of the algorithm, while **Lin** is closer to the FGIF algorithm in running efficiency, but much lower than the FGIF algorithm in fusion effectiveness. Considering both effectiveness and efficiency, the FGIF algorithm is much better than the other algorithms.

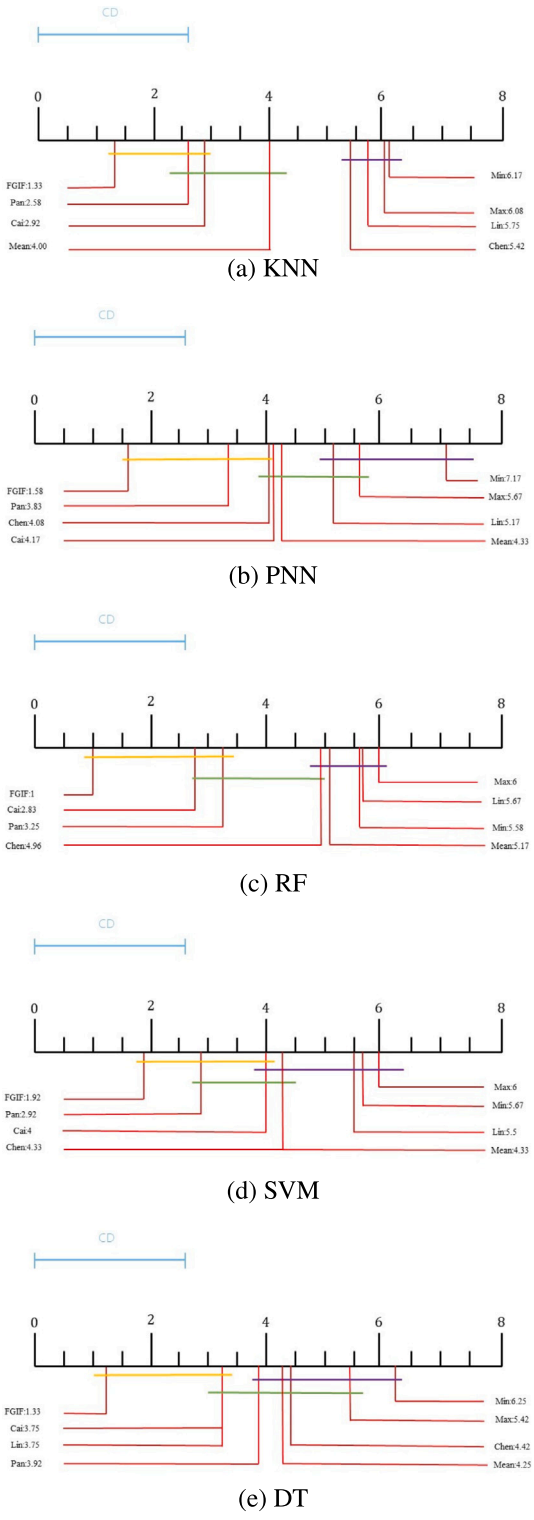

Fig. 8. The comparison of different algorithms under the Friedman test: (a)KNN classifier, (b)PNN classifier, (c)RF classifier, (d)SVM classifier, (e)DT classifier.

Table 11
The runtime(s) comparison with other algorithms on all datasets.

Dataset	FGIF	Lin	Pan	Chen	Cai
Dataset1	**4.8699**	8.1529	53.3829	76.994	4.9271
Dataset2	**0.1661**	0.7353	0.8225	0.646	0.7024
Dataset3	**1.4662**	1.5895	2.7841	3.023	2.0254
Dataset4	**40.131**	62.537	652.6393	667.376	632.507
Dataset5	**0.6057**	0.9497	2.4432	1.0802	1.9143
Dataset6	**1.3015**	1.4693	4.8097	2.5014	7.1646
Dataset7	**1.9241**	2.0373	18.1267	9.3776	41.6208
Dataset8	**4.8735**	5.9341	231.618	91.5042	674.626
Dataset9	99.4812	**92.3773**	37,086.1612	12,380.339	58,774.519
Dataset10	**226.8489**	239.091	1,188,510.811	89,892.52	4,706,532.63
Dataset11	**357.9939**	361.228	342,052.378	125,133.965	538,047.367
Dataset12	**898.3437**	947.089	1,059,791.934	627,457.161	2,769,536.866

5. Conclusion

In the introduction part of this paper, this study has illustrated that there are still shortcomings in the existing feature selection or information fusion framework based on rough sets: The computational complexity of the sample space distance in the neighborhood rough set is $O(n^2)$, which is inevitable. Meanwhile, the existing fusion algorithms for multi-source order decision information systems are relatively few. Moreover, since distance calculation is necessary in the acquisition of neighborhood rough sets, this also leads to the fact that most algorithms based on rough sets are sensitive to noise.

In this paper, an efficient graph-driven information fusion algorithm framework based on dominant rectangular granularity for MS-ODIS is developed. The main contributions of this algorithm are: i) reducing the time cost of calculating rough sets significantly, ii) proposing a novel information fusion framework by utilizing graph theory, and iii) demonstrating that the algorithm of neighborhood rough set in this paper which essentially uses the distance relationship between samples, is not sensitive to noise thereby providing a possibility to handle noisy data in the real world. For novelty, specifically, it involves replacing the calculation of the real space distance of samples in the traditional neighborhood rough set method, with the calculation of the rectangular neighborhood rough set. This is achieved by using the distance and introducing the adjacency matrix of the graph to aggregate the distance information between all samples thereby evaluating the dominance relationship of attributes more accurately. In the experimental part, the results are compared with other existing algorithms for effectiveness and efficiency across 12 datasets. The results of the experiments also provide evidence that the FGIF algorithm proposed in this paper achieves high operational efficiency while maintaining effectiveness compared with other algorithms.

However, there are still two improvements in this paper: i) Firstly, although the time complexity of computing the distance between samples is reduced from $O(n^2)$ to $O(nlogn)$ in this paper, the essence is that it trades space for time. We need to maintain a balanced binary tree in the cache to ensure that it can be queried at any time. ii) The flexibility of rough set granularity calculation is limited. In this algorithm, the distance r between samples is fixed to an integer which is determined by the structure of the balanced binary tree. This approach gives up the unbounded neighborhood radius in the traditional neighborhood rough set calculation. This limitation greatly reduces the possibility and diversity of coarse or fine granularity.

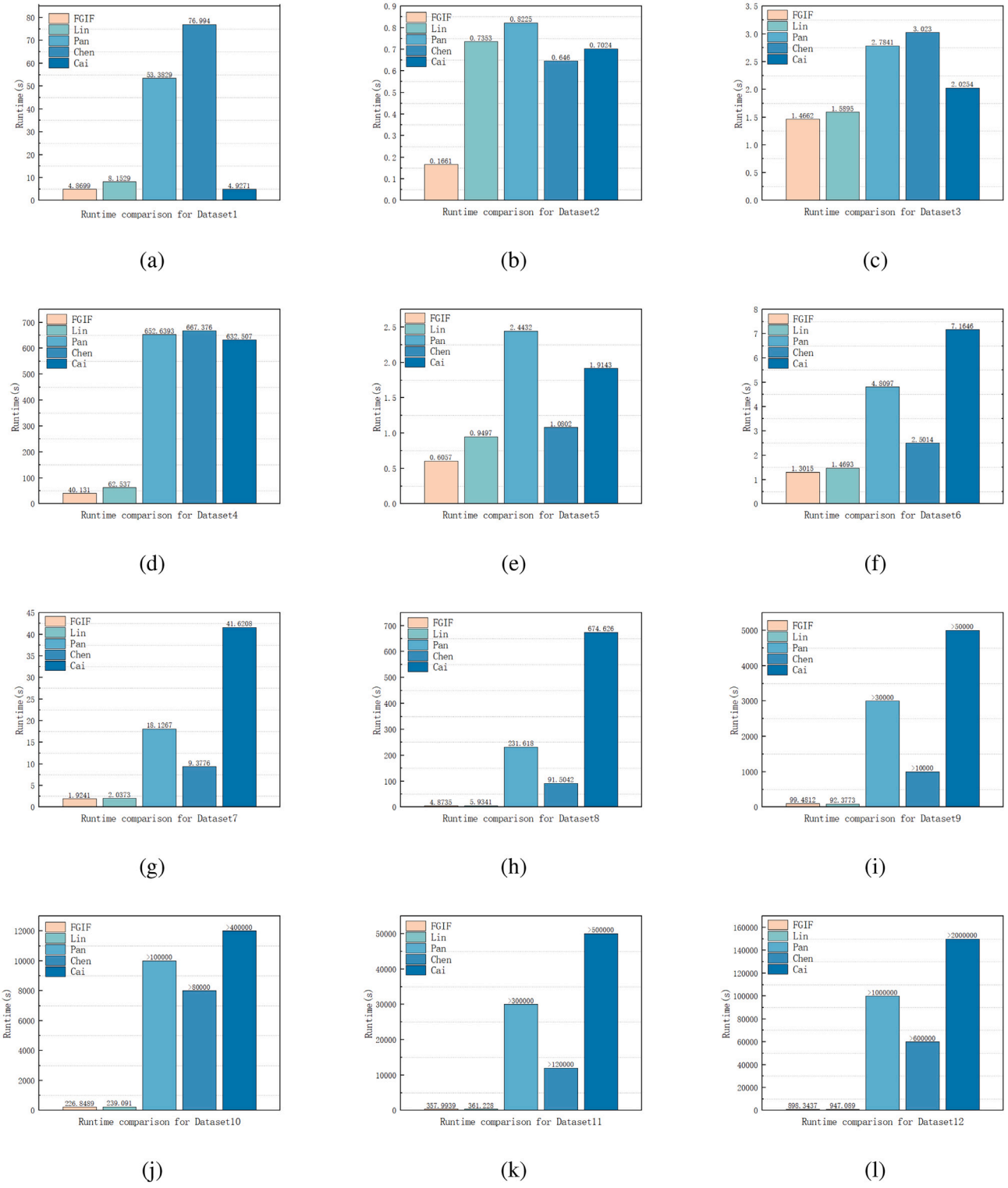

Fig. 9. The runtime comparison with other algorithms: (a)Tox, (b)Wine, (c)BCW, (d)MHR, (e)QSAR, (f)Wir, (g)Aba, (h)Ele, (i)Stu, (j)DHI, (k)Ind, (l)Cre.

CRediT authorship contribution statement

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

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 62376229) and Natural Science Foundation of Chongqing (No. CSTB2023NSCQ-LZX0027).

Data availability

The link to the data used for the research described has been shared in the article.

References

[1] Z. Liu, Y. Cao, X. Yang, L. Liu, A new uncertainty measure via belief Rényi entropy in Dempster-Shafer theory and its application to decision making, Commun. Stat. Theory Methods 53 (19) (2024) 6852–6868.
[2] K. Yuan, D. Miao, W. Pedrycz, W. Ding, H. Zhang, Ze-HFS: zentropy-based uncertainty measure for heterogeneous feature selection and knowledge discovery, IEEE Trans. Knowl. Data Eng. 36 (2024).
[3] Q. Wei, H. Zhang, Y. Chen, Y. Xie, H. Yin, Z. Xu, City scale urban flooding risk assessment using multi-source data and machine learning approach, J. Hydrol. 651 (2025) 132626.
[4] A. Uzzaman, Federated learning–driven real-time disease surveillance for smart hospitals using multi-source heterogeneous healthcare data, ASRC Procedia: Glob. Perspect. Sci. Scholarsh. 1 (1) (2025) 1390–1423.
[5] T. Wang, R. Liu, G. Qi, Multi-classification assessment of bank personal credit risk based on multi-source information fusion, Expert Syst. Appl. 191 (2022) 116236.
[6] P.A. Gbadega, Y. Sun, O.A. Balogun, Optimized energy management in grid-connected microgrids leveraging *K*-means clustering algorithm and artificial neural network models, Energy Convers. Manag. 336 (2025) 119868.
[7] Z. Pawlak, Rough set theory and its applications to data analysis, Cybern. Syst. 29 (7) (1998) 661–688.
[8] R. Abu-Gdairi, R. Mareay, M. Badr, On multi-granulation rough sets with its applications, Comput. Mater. Contin. 79 (1) (2024).
[9] A. Gaeta, V. Loia, F. Orciuoli, An explainable prediction method based on fuzzy rough sets, TOPSIS and hexagons of opposition: applications to the analysis of information disorder, Inf. Sci. 659 (2024) 120050.
[10] J. Liu, B. Sun, J. Ye, X. Zhao, X. Chu, Hybrid deep-learning prediction model based on kernel multi-granularity fuzzy rough sets and its application in the diagnosis and treatment of chronic kidney disease, Eng. Appl. Artif. Intell. 147 (2025) 110297.
[11] C. Ren, P. Zhu, Attribute reduction based on interval-set rough sets, Soft Comput. 28 (3) (2024) 1893–1908.
[12] X. Li, L. Li, C. Jia, Application of a novel multi-granularity variable precision fuzzy rough set in attribute reduction, Int. J. Fuzzy Syst. (2025) 1–19.
[13] X. Zhang, Q. Ou, J. Wang, Variable precision fuzzy rough sets based on overlap functions with application to tumor classification, Inf. Sci. 666 (2024) 120451.
[14] Y. Zhang, C. Wang, Y. Huang, W. Ding, Y. Qian, Adaptive relative fuzzy rough learning for classification, IEEE Trans. Fuzzy Syst. 32 (11) (2024) 6267–6276.
[15] W. Xu, K. Yuan, W. Li, W. Ding, An emerging fuzzy feature selection method using composite entropy-based uncertainty measure and data distribution, IEEE Trans. Emerg. Top. Comput. Intell. 7 (1) (2022) 76–88.
[16] J. Yang, Z. Liu, G. Wang, Q. Zhang, S. Xia, D. Wu, Y. Liu, Constructing three-way decision with fuzzy granular-ball rough sets based on uncertainty invariance, IEEE Trans. Fuzzy Syst. 33 (6) (2025) 1781–1792.
[17] X. Li, F. Dunkin, J. Dezert, Multi-source information fusion: progress and future, Chin. J. Aeronaut. 37 (7) (2024) 24–58.
[18] P. Zhang, T. Li, G. Wang, C. Luo, H. Chen, J. Zhang, D. Wang, Z. Yu, Multi-source information fusion based on rough set theory: a review, Inf. Fusion 68 (2021) 85–117.
[19] B. Sang, L. Yang, W. Xu, H. Chen, T. Li, W. Li, VCOS: multi-scale information fusion to feature selection using fuzzy rough combination entropy, Inf. Fusion 117 (2025) 102901.
[20] K. Cai, W. Xu, An efficient multi-source information fusion approach for dynamic interval-valued data via fuzzy approximate conditional entropy, Int. J. Mach. Learn. Cybern. (2024) 1–27.
[21] H. Yuan, W. Xu, A novel method of feature selection and information fusion for multi-source ordered information systems based on k-nearest neighbor rough sets, Inf. Sci. (2025) 122997.
[22] W. Xu, Y. Pan, X. Chen, W. Ding, Y. Qian, A novel dynamic fusion approach using information entropy for interval-valued ordered datasets, IEEE Trans. Big Data 9 (3) (2022) 845–859.
[23] Q. Kong, C. Yan, W. Xu, Simplified rough sets, Inf. Sci. 686 (2025) 121367.
[24] S. Wu, L. Wang, S. Ge, Z. Hao, Y. Liu, Neighborhood rough set with neighborhood equivalence relation for feature selection, Knowl. Inf. Syst. 66 (3) (2024) 1833–1859.
[25] C. Wang, C. Wang, Y. Qian, Q. Leng, Feature selection based on weighted fuzzy rough sets, IEEE Trans. Fuzzy Syst. 32 (2024).
[26] W. Xu, Y. Zhang, Y. Qian, A novel unsupervised feature selection for high-dimensional data based on FCM and *k*-nearest neighbor rough sets, IEEE Trans. Neural Netw. Learn. Syst. 36 (2024).
[27] W. Xu, J. Li, Granular-ball-matrix-based incremental semi-supervised feature selection approach to high-dimensional variation using neighbourhood discernibility degree for ordered partially labelled dataset, Appl. Intell. 55 (4) (2025) 1–26.
[28] W. Xu, Y. Lin, N. Wang, A novel multi-source information fusion method based on dependency interval, IEEE Trans. Emerg. Top. Comput. Intell. 8 (2024).
[29] P. Zhang, T. Li, Z. Yuan, C. Luo, G. Wang, J. Liu, S. Du, A data-level fusion model for unsupervised attribute selection in multi-source homogeneous data, Inf. Fusion 80 (2022) 87–103.
[30] W. Xu, K. Cai, D.D. Wang, A novel information fusion method using improved entropy measure in multi-source incomplete interval-valued datasets, Int. J. Approx. Reason. 164 (2024) 109081.
[31] X. Zhang, X. Chen, W. Xu, W. Ding, Dynamic information fusion in multi-source incomplete interval-valued information system with variation of information sources and attributes, Inf. Sci. 608 (2022) 1–27.
[32] A. Yolcu, A. Benek, T.Y. Öztürk, A new approach to neutrosophic soft rough sets, Knowl. Inf. Syst. 65 (5) (2023) 2043–2060.
[33] T. Shaheen, H.B. Khan, W. Ali, S. Najam, M.Z. Uddin, M.M. Hassan, A novel generalization of sequential decision-theoretic rough set model and its application, Heliyon 10 (13) (2024).
[34] J. Xu, C. Zhou, S. Xu, L. Zhang, Z. Han, Feature selection based on multi-perspective entropy of mixing uncertainty measure in variable-granularity rough set, Appl. Intell. 54 (1) (2024) 147–168.
[35] B. Sang, H. Chen, T. Li, W. Xu, H. Yu, Incremental approaches for heterogeneous feature selection in dynamic ordered data, Inf. Sci. 541 (2020) 475–501.
[36] Z. Feng, X. Zhang, Supervised incremental feature selection using regularization vector for dynamic multi-scale interval valued datasets, Pattern Recognit. 170 (2026) 111985.
[37] S. Xia, S. Wu, X. Chen, G. Wang, X. Gao, Q. Zhang, E. Giem, Z. Chen, GRRS: accurate and efficient neighborhood rough set for feature selection, IEEE Trans. Knowl. Data Eng. 35 (9) (2022) 9281–9294.
[38] X. Zhang, X. Shen, Graph-driven feature selection via granular-rectangular neighborhood rough sets for interval-valued data sets, Appl. Soft Comput. (2025) 112716.
[39] G. Roffo, S. Melzi, U. Castellani, A. Vinciarelli, M. Cristani, Infinite feature selection: a graph-based feature filtering approach, IEEE Trans. Pattern Anal. Mach. Intell. 43 (12) (2020) 4396–4410.
[40] Y. Fan, J. Liu, J. Tang, P. Liu, Y. Lin, Y. Du, Learning correlation information for multi-label feature selection, Pattern Recognit. 145 (2024) 109899.
[41] F. Nie, W. Zhu, X. Li, Structured graph optimization for unsupervised feature selection, IEEE Trans. Knowl. Data Eng. 33 (3) (2021) 1210–1222.
[42] Y. Akhiat, Y. Asnaoui, M. Chahhou, A. Zinedine, A new graph feature selection approach, in: 2020 6th IEEE Congress on Information Science and Technology (CiSt), IEEE, 2021, pp. 156–161.
[43] C. Tang, X. Zheng, W. Zhang, X. Liu, X. Zhu, E. Zhu, Unsupervised feature selection via multiple graph fusion and feature weight learning, Sci. China Inf. Sci. 66 (5) (2023) 152101.
[44] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, vol. 9, Springer Science & Business Media, 2012.
[45] Y. Huang, T. Li, C. Luo, H. Fujita, S.-J. Horng, Dynamic fusion of multisource interval-valued data by fuzzy granulation, IEEE Trans. Fuzzy Syst. 26 (6) (2018) 3403–3417.
[46] Q. Hu, D. Yu, J. Liu, C. Wu, Neighborhood rough set based heterogeneous feature subset selection, Inf. Sci. 178 (18) (2008) 3577–3594.
[47] G. Bierman, Power series evaluation of transition and covariance matrices, IEEE Trans. Autom. Control 17 (2) (1972) 228–232.
[48] R.L. Graham, Concrete Mathematics: A Foundation for Computer Science, Pearson Education India, 1994.
[49] E. Elizalde, Ten Physical Applications of Spectral Zeta Functions, Springer, 2012.
[50] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.