UHGSFS_1_

WORD COUNT 11319
TIME SUBMITTED 06-DEC-2023 08:21PM
PAPER ID 105118930

Online Unsupervised Heterogeneous Group Streaming Feature Selection

Peng Zhou[a,b,c], Qianzhen Chen[a,b,c], Lei Sang[a,b,c], Shu Zhao[a,b,c]

[a] *Key Laboratory of Intelligent Computing and Signal Processing, Ministry of Education, Anhui Province 230601, P.R. China*

[b] *School of Computer Science and Technology, Anhui University, Hefei, Anhui Province 230601, P.R. China*

[c] *Information Materials and Intelligent Sensing Laboratory of Anhui Province, Anhui Province 230601, P.R. China*

Abstract

Online streaming feature selection emerges as a preeminent technique for extracting optimal feature subsets from high-dimensional datasets, while traditional feature selection methods are inefficient for vast volumes of data. In practical applications, there is a large amount of unlabeled data, and features may be heterogeneous and be generated in streaming groups. Besides, since streaming features arrive randomly, predicting feature type information for the following arriving streaming features in advance is impractical. Most existing supervised streaming feature selection methods need labeled data for training. Meanwhile, existing unsupervised streaming feature selection techniques have the presumption of feature homogeneity and an inability to accommodate unknown feature types. To solve this practical problem, we propose a new online unsupervised heterogeneous group streaming feature selection method named UHGSFS. Based on maximum information coefficient and adaptive clustering techniques, UHGSFS does not require parameter setting for streaming feature clustering and utilizes feature redundancy minimization to select the most representative features. Extensive experimentation was conducted on 14 benchmark datasets, complemented by comparative analyses against state-of-the-art supervised and unsupervised streaming feature selection algorithms. Experimental results demonstrate the effectiveness and practicality of our new method.

Keywords: Unsupervised Feature Selection, Streaming Feature, Unknown Feature Type, Maximal Information Coefficient

Email addresses: doodzhou@ahu.edu.cn (Peng Zhou), qianzhen@stu.ahu.edu.cn (Qianzhen Chen), sanglei@ahu.edu.cn (Lei Sang), zhaoshuzs@ahu.edu.cn (Shu Zhao)

1. Introduction

The primary objective of feature selection is to discern optimal subsets from the original feature set, thereby enhancing model performance [1]. In the contemporary era of big data, the exponential proliferation of data has precipitated the advent of expansive feature space, thereby engendering what is commonly referred to as the "dimensionality disaster" [2]. Feature selection emerges as a pivotal instrument to address this predicament. Through the choosing and curation of pertinent features and the discarding of extraneous ones, feature selection not only refines learning performance but also engenders heightened computational efficiency and diminished memory storage requirements [3, 4].

Figure 1: Illustration of a real-world scenario for unsupervised heterogeneous group streaming feature selection. When a new patient enters the hospital, several inspection items are required to determine the cause of the illness. The results (features) of these different inspection items arrived randomly over time, resulting in groups of heterogeneous streaming features. How to deal with the heterogeneous streaming feature groups without labeling information is the problem we want to investigate.

With the increase in data volume and dimensionality, traditional feature selection methods can no longer meet the demand in terms of efficiency [2]. Meanwhile, traditional feature selection methods assume that all instances and features in the target dataset can be required before learning [5]. However, in real application scenarios, e.g., in image analysis [6], Mars exploration [7], and healthcare [8], we are often confronted with streaming features. As shown in Fig.1, a patient needs to be examined by different items, and different features are constantly generated by different examination items. The task of online streaming feature selection is to dynamically process features that keep arriving over time while the number of instances remains fixed [9]. Online streaming feature selection faces two challenges: 1) The size of the entire feature space is unknown or even infinite; 2) The algorithm needs to decide whether to keep or discard newly arrived features since

storing all these streaming features is impossible [10].

In recent years, many techniques for supervised streaming feature selection have been proposed, such as Fast-OSFS [9], OFS-Density [11], OFS-3WD [10], OMSFSLC [12] and LOSSA [13]. They mainly contain these four steps: 1) Require new streaming features (single/group); 2) Determine the new features to be added to the selected subset or discarded directly by analyzing the feature relevance; 3) Update the selected feature subset by redundancy analysis; 4. Repeat steps 1 to 3 until all the features have been examined. Although considerable progress has been made in supervised streaming feature selection, the reliance on data labeling makes these algorithms unsuitable for realistic scenarios with large amounts of unlabeled data.

On the other hand, unsupervised streaming feature selection needs to be studied more. In [14], an unsupervised streaming feature selection approach (named USFS) has been introduced for analyzing social media data using link information. USFS is only applicable to isomorphic data and has high computational complexity. In [15], Unsupervised Feature Selection for Dynamic Features (named UFSSF) extends k-means clustering by clustering continuous streaming features at the individual level. However, the performance of k-means clustering is sensitive to noise and is not applicable to handle discrete features. [16] proposed an online unsupervised streaming feature selection framework based on dynamic density feature clustering, but it needs to know the type of streaming features (discrete or continuous) in advance. [17] proposed a new feature selection method for feature streaming, which introduces a dynamic similarity graph to evaluate irrelevant features adaptively and utilizes similarity graph diffusion to eliminate those irrelevant features in the feature streams. The current state-of-the-art unsupervised streaming feature selection methods exhibit significant computational complexity, are constrained to isomorphic features, and cannot effectively handle heterogeneous streaming features. Besides, the order of arrival of streaming features is random in practical. Therefore, it is unrealistic to know the feature type of the next arriving streaming features in advance.

For example, in a real-world application scenario (shown in Fig.1), a new patient needs various medical diagnostic procedures in sequence to determine the cause of the disease. Over time, these inspection items will continue to generate various streaming data, such as image data from X-rays, signal data from ECGs, and text data from electronic case reports. The influx of large amounts of streaming data makes applying feature selection imperative. This issue has three challenges: 1) Due to the lack of labeled information related to new patients, supervised streaming feature selection

methods that rely on labeled data become inapplicable; 2) Streaming features are continuously generated, and the feature selection process should be applied on the fly; 3) The feature types of each group streaming features are heterogeneous. Moreover, the arrival of the next streaming group is random. In other words, we cannot know the feature type of the next streaming group in advance. Most existing unsupervised feature selection methods, which rely on the prerequisites of feature isomorphism and feature type prediction, are insufficient to cope with this intricate situation.

Motivated by this, we investigate a new unsupervised online heterogeneous streaming feature selection method where the streaming features are heterogeneous and dynamically generated in groups. Besides, we assume that the feature types of streaming features are unknown in advance for practical situations. Specifically, we use MIC (Maximum Information Coefficient) [18], which can discover nonlinear relationships and complex correlation structures to measure the relationship between heterogeneous features. Then, we utilize feature streaming density analysis to compute the density of each feature. Based on this, we propose a new adaptive feature clustering to group features with high redundancy. Finally, we perform the feature selection by selecting the feature with the highest density for the streaming feature selection. Our new method satisfies feature relevance maximization and feature redundancy minimization. The main contributions of this paper are as follows:

- We first focus on the practical issue of online unsupervised heterogeneous streaming feature selection where streaming features are generated dynamically in groups without the information of feature types in advance.
- We propose a new unsupervised heterogeneous group streaming feature selection method based on density-based streaming feature clustering. Specifically, we first evaluate the density of arriving features through MIC and feature streaming density formulas. We then employ a novel adaptive feature clustering method that does not require parameter setting for feature clustering and utilizes feature redundancy minimization to select the most representative subset of features.
- Extensive experiments were conducted on 14 datasets, involving a meticulous examination and comparison with eight online supervised streaming feature selection methods and one online unsupervised streaming feature selection algorithm. The experimental results, coupled with

rigorous statistical analyses, conclusively indicate that, in the absence of labeling information and under the circumstance of unknown streaming feature types, the performance of our new method is tantamount to that of supervised streaming feature selection algorithms or even better.

The rest of this article is organized as follows. In Section 2, we describe related work. In Section 3, the formal definition of the problem, the technology involved, and a new approach to unsupervised heterogeneous streaming Feature Selection is proposed. Section 4 gives the experimental analysis. Finally, Section 5 gives a brief conclusion.

2. Related Work

Research on feature selection has been conducted for many years, and a large number of excellent algorithms have been proposed [5]. According to different types of data generation, we can classify feature selection into two categories: traditional feature selection for static data and online feature selection for streaming data [2].

2.1. Traditional Feature Selection Methods

Depending on how the labeling information is used, traditional feature selection algorithms can be classified as supervised algorithms, unsupervised, and semi-supervised algorithms.

Supervised feature selection assumes we can require all the label information for the training data. More specifically, Fisher Score [19] achieved the same eigenvalues in the same class and different eigenvalues in different classes by calculating the ratio of interclass separation and intraclass variance for each feature. ReliefF [20] was a feature selection method based on the distance between samples, which performs feature selection by calculating the importance of each feature for sample classification. Mutual Information can measure the correlation between each feature and the target variable, and [21] discussed the practical implementation of mutual information (MI) in feature selection.

Without the labeling information, unsupervised feature selection cannot use the label to measure the importance of each feature. For example, Laplacian Score [22] evaluated the importance of features by their variance and power of locality preserving and matches similar features by constructing a nearest neighbor graph. FSFS [23] introduced the Maximum Information Compression

Index (MICI) to reduce feature redundancy by using clustering features with high similarity and selecting the most compact feature in each cluster (determined by the distance between features). MCFS [24] processed feature selection by exploring the clustering structure of the data and then using regularized regression to measure the importance of the features and ultimately selecting the features with the most informative value. SPLR [25] was an unsupervised feature selection method that combines self-paced learning into a subspace learning framework. SPLR preserved global data reconstruction information while effectively reducing the adverse effects of outliers. SDFS [26] was a multi-view unsupervised feature selection based on structural regularisation. SDFS calculated the similarity matrix of sample space from different views and automatically weighted each view-specific graph to learn a consensus similarity graph.

Besides, Semi-supervised feature selection assumes that we have a small amount of labeled data and a large amount of unlabelled data. For instance, SRFS [27] was a semi-supervised feature selection method that treats unlabelled data in a Markov blanket as labeled data through relevance gain. A-SFS [28] was a semi-supervised feature selection based on multi-task self-supervision, which innovatively introduces a self-supervision mechanism based on deep learning into the feature selection problem.

In summary, traditional feature selection methods necessitate full knowledge of the feature space before learning, rendering them unsuitable for online scenarios. Moreover, the substantial volume of data often results in computational overhead and time complexity that traditional feature selection methods, when applied to large-scale datasets, frequently fail to meet real-time requirements.

2.2. Online Streaming Feature Selection Methods

2.2.1. Supervised Streaming Feature Selection

With the advent of the big data era, online streaming feature selection has been a research hotspot dealing with high-dimensional datasets. In general, online supervised streaming feature selection can be divided into individual streaming feature selection and group streaming feature selection.

Individual streaming feature selection assumes features arrive one by one over time. For example, Grafting [29], which employs a stagewise gradient descent approach, represented the first individual-level streaming feature selection (SFS) method. Grafting considered feature selection as an integral part of learning a predictor within a regularized framework. Zhou et al. [30] pro-

posed the Alpha-investing algorithm, which leverages streamwise regression for online streaming feature selection. Specifically, Alpha-investing is utilized as penalized likelihood ratios in streamwise regression for streaming feature selection (SFS). Based on the redundancy minimization and relevance maximization processes, the online streaming feature selection (OSFS) framework and its named Fast-OSFS [9] were proposed. The Scalable and Accurate Online Feature Selection Approach (SAOLA) [31] maintained a parsimonious model by using a pairwise comparison method based on mutual information theory. OMSFSLC [12] investigated multi-label streaming feature selection in real-life scenarios where labels are interdependent and interrelated, and the method evaluates the relevance of features to labels through mutual information and correlation between features and labels. OFS-3WD [10] was a new online scalable streaming feature selection framework from a dynamic decision-making perspective, which dynamically classifies input features as selected, discarded, or delayed to minimize decision risk. OCFSSFs [32] was an online feature selection method based on causal discovery by mining and identifying relationships in Markov blankets about parents and children (PCs) and spouses. LOSSA [13] was an online sparse flow feature selection algorithm based on latent factor analysis, which uses latent factor analysis to solve the problem of missing data in sparse flow features. Meanwhile, some Rough Set-based SFS methods were proposed recently, including K-OFSD [33], OFS-A3M [34], OFS-density [11], and ASFS [35].

Existing supervised methods have achieved good performance in different aspects. However, these methods cannot be applied to unsupervised application scenarios since they require labeled data for training.

2.2.2. Unsupervised Streaming Feature Selection

In practice, data is mostly unlabelled. Online unsupervised streaming feature selection methods were proposed to handle online feature selection without label information.

Specifically, in [14], the authors performed unsupervised streaming feature selection against social media data by identifying link information. This method is only applicable to social media data. Meanwhile, it can only handle isomorphic feature streaming and requires stable link information with high computational complexity. In [15], the method implemented unsupervised streaming feature selection based on k-means for continuous individual feature streams, which cannot handle discrete features because k-means is sensitive to noise. In [16], the authors developed a feature density formulation adapted to streaming features and achieved feature correlation maximization and

redundancy minimization based on density clustering. However, it can not handle feature selection for mixed streaming data and artificially sets a parameter for clustering, which is not in line with real-world applications' unsupervised and unparameterized nature. [17] proposed a new feature selection method for feature streaming, which introduces dynamic similarity graphs to adaptively evaluate irrelevant features and uses similarity graph diffusion to eliminate those irrelevant features in the feature streaming.

In sum, most existing streaming feature selection approaches are constructed upon supervised information, which overlook the scarcity of labels in practical applications. While some researchers have introduced unsupervised streaming feature selection methods, their high computational complexity, restriction to isomorphic features, and incapacity to handle unknown features render these methods unsuitable for addressing unsupervised heterogeneous streaming feature selection challenges.

3. The Proposed Framework

This section describes the definition of the problem and the concrete implementation of the proposed method. We summarize some of the notations used in this paper in Table 1.

3.1. Notations, Assumptions, and Definitions

Let $\mathbb{F} = \{G_1, G_2, ..., G_T\}$ representes the entire streaming features, where $G_t = \{f^1_{G_t}, f^2_{G_t}, ..., f^m_{G_t}\}$ denotes the streaming feature group arriving at timestamp t. For each streaming group G_t, it is heterogeneous, and we cannot know the feature type information for each streaming feature $f^i_{G_t}$ in it. At each timestamp t, FS_t denotes the selected feature subset by the algorithm. Online unsupervised streaming feature selection aims to obtain an optimal feature subset FS_t from $\mathbb{F}$ under the following assumptions:

- The quantities of samples and classes are fixed;
- Streaming features arrive as groups over time;
- The label and feature type information for each streaming group are unknown.

Since feature space changes over time, we cannot store all features for online streaming feature selection. Meanwhile, since there is no labeling information, we have no basis for judging the importance of features. Therefore, unsupervised streaming feature selection often requires discovering

Table 1: Summary on Mathematical Notations

Notations	Definition
t	the timestamp
n	total number of samples
G_t	a group of streaming features arrive at timestamp t
T	the total number of streaming feature groups
m	the number of the features in the group G_t
f_i^i	the i^{th} streaming feature in G_t
FS_t	the selected feature subset after timestamp t
D_{f_i}	the density value of streaming feature f_i
FC_t	the extracted streaming feature cluster set at timestamp t
C	a cluster of streaming features in FC_t
r_C	the radius of the cluster C
$\|FC_t\|$	the number of feature clusters in FC_t
$d(f_a, f_b)$	the distance from the feature f_a to the feature f_b

relationships between features to select or discard features. In addition, we need an effective metric to explore the relationships between heterogeneous streaming features.

To solve this problem, we apply density-based streaming feature clustering to select the representative features and discard the redundant features. Suppose the streaming features are clustered into multiple clusters FC_t at timestamp t, and C is one of cluster in FC_t.

Definition 1. **[Representative feature]** The representative feature in streaming feature cluster C is defined as the feature with the maximal density as: $f_R = \underset{f \in C}{max}\{D_f\}$, where D_f is the density value of streaming feature f.

For the new arriving streaming group G_t, we calculate the density of each streaming feature in it and cluster them into multiple clusters. We designate the feature by the local maximum density value as the cluster center and the representative feature.

Definition 2. **[Redundant features]** For the features in the same cluster C, features other than representative features are considered redundant.

Based on steaming feature clustering, highly correlated features are clustered in the same cluster. According to **Definition 1**, we selectively choose the most representative feature from each cluster and consider the other features redundant. This approach can effectively reduce feature redundancy.

Based on these two definitions, our online unsupervised streaming feature selection method can effectively implement a strategy of maximizing relevance and minimizing redundancy.

3.2. Our New Framework

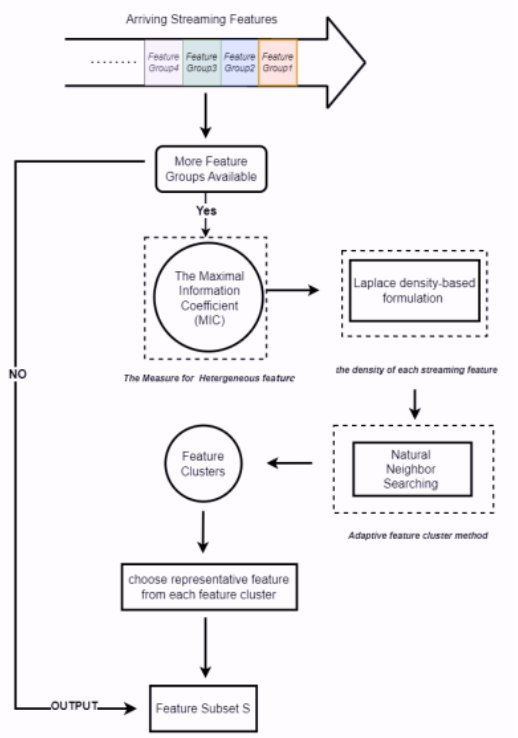


Figure 2: An overview of the Unsupervised Heterogeneous Group Streaming Feature Selection framework.

The depicted framework of our proposed Unsupervised Heterogeneous Group Streaming Feature Selection (UHGSFS) is illustrated in Fig.2. Specifically, we first apply the Maximum Information Coefficient (MIC) to measure the information between heterogeneous streaming features, which can discover complex linear relationships for a distance evaluation. Then, we calculate the density of each streaming feature. We select the feature with the current highest density to perform adaptive feature clustering to cluster the features into different clusters. Finally, we obtain the optimal feature subset by selecting the most relevant features and discarding redundant features.

3.2.1. The Measure for Unknown Type and Heterogeneous Feature

The generated streaming features in practical applications often exhibit heterogeneity. Meanwhile, each streaming feature arrives randomly, and we cannot know their feature types in advance. Previous streaming feature selection methods, supervised or unsupervised, have predominantly been designed to handle a single feature type (discrete or continuous). Therefore, there is an urgent need to develop novel metrics that can effectively address the challenges posed by heterogeneous streaming features of unknown types.

MIC (maximal information coefficient) is a data analysis algorithm that evaluates variable relationships without assuming data distribution [18]. MIC partitions a scatter plot into grids and computes dissimilarity using mutual information. The highest mutual information value represents the Maximum Information Coefficient (MIC). MIC has been proven to effectively measure the dependence of two variables and can capture a wide range of functional and unfunctional associations.

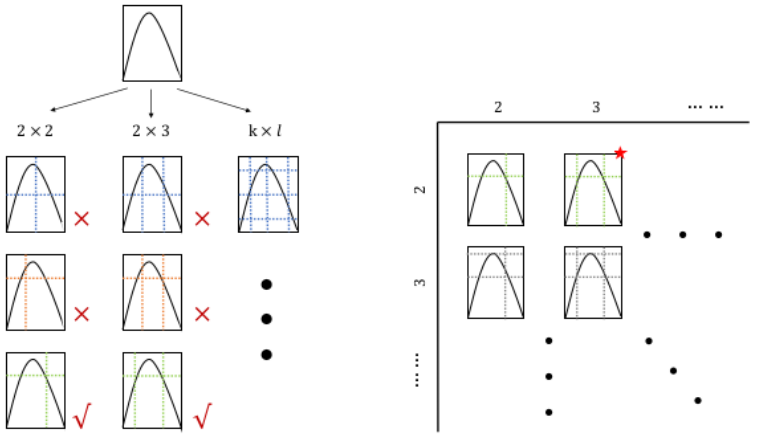
In Fig.3, MIC employs dynamic axis division, enabling the calculation of mutual information for both numerical and categorical data, rendering it highly versatile across diverse applications. For a two-dimensional variable dataset $D = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, the parameters (k, l) may assume any valid pair. The calculation of the MIC(D) is as follows:

$$MIC(D) = max\{M(D)_{k,l}\} \tag{1}$$

$$M(D)_{k,l} = \frac{maxMI(D)_{(k,l)}}{log(min\{k, l\})} \tag{2}$$

Where $MI(D)_{(k,l)}$ denotes the mutual information value divided according to the integers (k, l) on the two-dimensional variable dataset D. The size of k and l when the party mutual information is the maximum value can be obtained by the exhaustive method. $k \times l \leq B(n)$, B is a function of the sample size n expressed as $B(n) = n^{0.6}$.

For online heterogeneous streaming feature selection, the inherent uncertainty lies in the inability to ascertain the feature type of the next streaming feature preemptively. The Maximum Information Coefficient (MIC) uniquely quantifies correlations across various types of variables. In light of our objective to compute feature distances, we leverage the concept that higher correlations result in shorter distances. Therefore, we incorporate the complementary metric of $1 - MIC$ to effectively measure and represent the distances between these heterogeneous streaming features within our



(a) original instances distribution

(b) new spatial instancesdistribution aftertransductive warping

Figure 3: Taking a parabola as an example, a schematic diagram of calculating MIC.

novel information metric framework.

3.2.2. The Density of Streaming Feature

Before we cluster these arriving streaming features in groups, we need to calculate the density of each feature. In [16], the authors introduced a pioneering two-phase streaming clustering methodology referred to as **dpps-clustering**. This approach treats the critical data processing task as an optimization process, demonstrating superior efficacy even when substantial prior knowledge is lacking. [16] derived a multilevel lower bound function using the Laplacian density formula and replaced it with clusters. The density distribution in the input feature group is calculated, and the derived Laplacian density lower bound effectively solves the problem of noise features in the streaming. The lower bound density function of each streaming feature is calculated as follows:

$$D_{f_i} = \sum_{f_j \in G_t} e^{-(\frac{d(f_i,f_j)}{\beta_t})^{\gamma_t}} + \sum_{f_k \in FC_{t-1}} e^{-(\frac{d(f_i,f_k)}{\beta_t})^{\gamma_t}} \times D_{f_k} \tag{3}$$

where f_i is the streaming feature in group G_t, $d(f_i, f_j)$ is the feature distance between features f_i and f_j. Besides, for each selected candidate feature f_k in FC_{t-1}, we calculate the distance between f_i and f_k. D_{f_k} is the density value of f_k. β_t and γ_t are the normalization parameters and stabilization parameters at timestamp t, respectively. We employ the identical parameter estimation procedure

outlined in [36] to derive β_t and γ_t. Meanwhile, we define $d(f_i, f_j) = 1 - MIC(f_i, f_j)$ as the distance function in this paper.

By applying this formula, we can effectively estimate feature density within the current feature group by leveraging the selected candidate features and the current streaming feature group. Significantly, the recursive lower bound derived from this formula relies exclusively on the retention of clustering historical data (selected features), eliminating the need to store the entirety of the historical streaming data.

3.2.3. Adaptive Feature Cluster Method

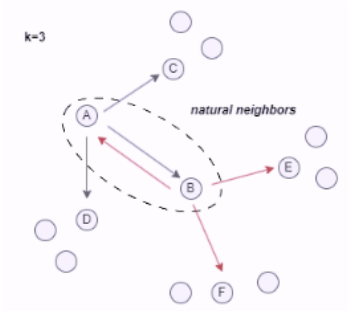


Figure 4: Natural neighborhood search. Suppose we are looking for k nearest neighbors (k=3). It can be observed that node A is closest to nodes B, C, and D. Meanwhile, node B is closest to nodes A, E, and F. Based on the natural neighbors search, it can be inferred that nodes A and B are neighbors.

"Natural Neighbors" introduces a novel paradigm of proximity akin to human social interactions: Take, for instance, a trio of individuals denoted as A, B, and C. If A lacks amicability towards B, while B reciprocates with friendliness towards A, one may infer that A's affinity towards B is unidirectional rather than a reciprocal friendship. Conversely, if A and C both display mutual animosity, it establishes that they do not share a close bond. In contrast, the amicability between B and C suggests a genuine friendship. Unlike traditional k-nearest neighbors search, Fig.4 illustrates the natural neighbor search process.

Diverging from the conventional k-nearest neighbor (KNN) approach, the natural nearest neighbor method leverages the intrinsic structure of the dataset to discern the closest neighbors for each data instance. Noteworthy is its unique capacity to dynamically ascertain the optimal number of nearest neighbors for each data point, obviating the need for a priori parameter specification.

Motivated by the notion of natural neighborhoods, we extend this concept to devise an adaptive feature clustering method tailored for feature streaming data.

Given a data set $\mathbb{X}$, we define $d(p,q)$ as the distance between data points p and q in $\mathbb{X}$. Let data point k be the k^{th} nearest neighbors of data point p. The interpretations of k-nearest neighbors, reversed k-nearest neighbors, natural stable state, and natural search neighbor are given as follows.

Definition 3. [**K-Nearest Neighbors**] **KNN**: For data point p, its k-Nearest Neighbors set can be expressed as:

$$KNN_k(p) = Find_{knn}(p,k) \tag{4}$$

where $Find_{knn}(p,k)$ represents the searching function of KNN which searches for the k^{th} nearest neighbors of p. Here, a KD tree can be utilized to accelerate the KNN searching process.

Definition 4. [**Stable Searching State**]: Suppose the searching round r increases from 1 to n. The natural neighbor searching becomes stable when the following condition is satisfied:

$$(\forall p \in \mathbb{X}) \wedge (\exists q \in \mathbb{X}) \wedge (p \neq q) \longrightarrow p \in KNN_r(q) \wedge q \in KNN_r(p) \tag{5}$$

When the stable searching state is reached, the maximal search round r is called the natural neighbor eigenvalue.

Definition 5. [**Natural Search Neighbors**]: When the natural neighbor searching process keeps a stable searching state, the data point p and data point q are natural neighbors if $p \in KNN_r(q)$ and $q \in KNN_r(p)$.

Motivated by the concept of natural neighbor search, we have formulated an algorithm for adaptive feature clustering, drawing upon the definition and analysis outlined above. A comprehensive exposition of the clustering process employing natural feature neighbor search is presented in Algorithm 1.

Specifically, for the target feature f, we aim to find its maximal natural neighbors as the cluster of f. Step 1 initializes $r = 1$. Then, we continuous search the r^{th} neighbor of f as q in Step 3. Steps 4-7 check whether f belongs to the r nearest neighbors of q. If it does, then r is incremented by 1. Otherwise, it means that feature f reaches the **STABLE SEARCHING STATE** and we end the searching.

After we have assigned all the features to the possible clusters by Algorithm 1, we determine whether there is a high degree of overlap between the clusters by the presence of a density valley

Algorithm 1 Natural Neighbor Clustering

Input:

F: Feature set;

f: Feature f;

Output:

C_f: The cluster of feature f;

Initialization: $r = 1$;
Repeat
    Find the r^{th} nearest neighbors of f as q
    If $f \in KNN_r(q)$
        $C_f = C_f \cup \{q\}$;
        r=r+1;
    End If
Until f reaches stable searching state
Return C_f

between the two clusters. Density valley [37] refers to a point between two cluster cores as follows:

Definition 6. [Density Valley]: Suppose f_B represents the boundary feature of cluster $C1$, f_{C1} and f_{C2} represent the cluster centers of the two clusters nearby, if $D_{f_B} < MIN(D_{f_{C1}}, D_{f_{C2}})$, it means that there exists a valley of density between these two clusters.

In other words, there is a density valley if the density of the boundary feature f_B is lower than the minimum of these two cluster cores. There is no need to merge if there is a density valley between these clusters. In contrast, if there is no density valley between the clusters, the two clusters highly overlap, so merging is required.

3.2.4. The Proposed Framework.

Since the online streaming feature selection operates unsupervised, it necessitates consideration of feature correlation and redundancy. Without data labels for measuring feature importance and relevance, we rely on the representation within each feature cluster to assess feature significance. In the context of density-based clustering methods, it is customary for the cluster center to exhibit the highest local density value, thus establishing the feature cluster center as the most pertinent feature within the cluster. Moreover, given that features with greater similarity are typically chosen within

feature clusters, it is reasonable to infer that features of the same cluster exhibit redundancy. By exclusively selecting feature centers, we can effectively minimize redundancy. Employing these two strategies enables the extraction of crucial feature subsets from a feature streaming while maintaining low redundancy and high correlation among them. Drawing from the preceding solution, we introduce UHGSFS, an unsupervised method designed to select heterogeneous streaming features, as shown in 2.

Algorithm 2 Unsupervised Heterogeneous Group Streaming Feature Selection(UHGSFS)

Input:

G_t: a group of streaming features arrive at timestamp t;

Output:

FS_t: the selected feature subset;

Repeat
  $D_{G_t} = \{\}$: the density values of each feature in G_t;
  For each feature $f^i_{G_t} \in G_t$
    Calculate $D_{f^i_{G_t}}$ using Eq.3 with MIC;
    $D_{G_t} = D_{G_t} \bigcup \{D_{f^i_{G_t}}\}$;
  End For
  Rank all features in G_t based on their density values;
  $FC_t = \{\}$: the extracted streaming feature cluster set;
  Repeat
    Select the feature f in G_t with the maximal density value
    Generate the cluster C_f by Algorithm 1;
    $FC_t = FC_t \cup C_f$;
    G_t=G_t-C_f;
  Until no more features in G_t
  Update FC_t by merging clusters based on possible density valleys;
  If FS_{t-1} in not empty
    Merge FC_t with historical feature clusters in FC_{t-1};
  End If
  FS_t=select representative feature from each cluster in FC_t;
Until no more streaming feature groups
Return FS_t

Specifically, suppose UHGSFS gets a new streaming feature group G_t at timestamp t. Steps 3-6 calculate the density values of each feature in G_t in terms of Eq.3 and the maximum information coefficient (MIC). In step 7, we sort all the features by their density values. Steps 9-14, we select feature f with the highest density in G_t at each time and apply Algorithm1 to generate the cluster of f until all the features are assigned to the clusters. Step 15 determines whether merging is required by the presence of density valleys between the two clusters. In steps 16-18, we check the possible merging of current clusters FC_t and historical clusters in FC_{t-1}. Finally, we select the representative feature from each cluster as the selected feature subset.

3.3. Time Complexity

In our algorithm, the computation of the feature distance matrix for the unknown heterogeneous feature streaming necessitates $O(m^2)$ distance calculations. The search for feature clusters through our natural neighbor search method incurs $O(nlogn)$ computational complexity. Likewise, the merging process involving historical feature clustering and new feature clustering entails $O(|FC_t||FC_{t-1}|)$ calculations. As a result, the worst-case time complexity of our algorithm is expressed as $O(m^2 + nlogn + |FC_t||FC_{t-1}|)$.

4. Experiments

In this section, we conduct experiments on several benchmark datasets and comparative studies with state-of-the-art supervised and unsupervised streaming feature selection methods to demonstrate the effectiveness of our new framework.

4.1. Experiment Setup

4.1.1. Datasets

To verify the effectiveness of our proposed new method, we conducted extensive experiments on 14 real-world datasets, including four discrete datasets and ten consecutive datasets. We can find these datasets from the ASU feature selection repository[1]. Table 2 summarizes the properties of these datasets in terms of sample size, feature size, class size, application domain, and type.

[1] Public available at https://jundongl.github.io/scikit-feature/datasets.html

Table 2: Real-world Datasets

Data Set	instances	Features	Classes	Domain	Type
Arcene	200	10000	2	Medical	continuous
AllMall	72	7129	2	Medical	continuous
LUNG	73	325	7	Medical	discrete
Lymphoma	96	4026	9	Medical	discrete
Carcinom	174	9182	11	Biological	continuous
Colon	62	2000	2	Biological	discrete
GLI-85	85	22283	2	Biological	continuous
GLIMA	50	4434	4	Biological	continuous
Nci-9	60	9712	9	Biological	discrete
SMK-CAN-187	187	19993	2	Biological	continuous
Coil20	1440	1024	40	Image	continuous
Orlraws10P	100	10304	10	Image	continuous
Pixraw10P	100	10000	10	Image	continuous
WarpPIE10P	210	1024	15	Image	continuous

4.1.2. Comparing Algorithms

We compare our new method with nine state-of-the-art streaming feature selection methods [1], including Alpha-investing [30], SAOLA [31], Fast-OSFS [9], OFS-Density [11], OFS-A3M [34], OFS-3WD [10], Group-SAOLA [31], OGSFS-FI [38] and OUFSDFC [16]. The first eight algorithms are supervised, and OUFSDFC is an unsupervised method. Meanwhile, the first six algorithms are individual streaming feature selection methods, while the last three are group-based approaches. According to [38], the α value of Fast-OSFS, SAOLA, and Group-SAOLA is set to 0.01. For Alpha-investing, OFS-density, OFS-A3M, and OGSFS-FI, the default parameter settings in [11, 34] are used to obtain the final selected feature subset.

4.1.3. Evaluation Metrics

Classification performance is used to show the effectiveness of these competing algorithms. Fundamental classifiers were employed, including KNN (k = 5), SVM (with the linear kernel), and DT(decision tree), to evaluate the feature subset selected in the experiment. For each dataset, we use a 5-fold cross-validation method, dividing the dataset into five equal parts, with four parts

[1] Public available at https://github.com/kuiy/LOFS, and https://github.com/doodzhou/OSFS.

for training and one part for testing each time. We utilize the same training and testing set division for each competing algorithm.

Two well-known evaluation metrics, including Accuracy (Acc) and f-score (F_{mac}), are used as performance evaluation metrics. To account for imbalances in class distribution, we use the macro-average of f-score, which is expressed as follows:

$$F_{mac} = \frac{1}{n_c} \sum_{i=1}^{n_c} F_i \tag{6}$$

where F_i and n_c denote the F-measure for the i^{th} class and the number of classes, respectively.

Besides, we conducted the statistical test using Friedman's test at a significance level of 95%, assuming the null hypothesis [39]. If the null hypothesis is rejected, we conduct the Nemenyi test as a post-hoc test and construct critical distance (CD) graphs [40].

4.1.4. Computational Device

All experiments were conducted on a computer running Windows 10, AMD Ryzen 7 3700X 8-core processor 3.6 GHz, 16 GB RAM.

4.2. Results and Discussions

Tables 3-8 present a comprehensive summary of the accuracy (ACC) and F_{mac} achieved by these competing algorithms, utilizing KNN, SVM, and DT classifiers. Additionally, Table 9 provides insights into each algorithm's average number of selected features.

UHGSFS and FOUFSDFC are implemented in Python, while the other eight algorithms are implemented in Matlab. Thus, we can only compare the running time between UHGSFS and OUFSDFC. Fig.6 compares the running times between UHGSFS and OUFSDFC algorithms.

For the accuracy metric (ACC), the p-values obtained from the Friedman test for KNN, SVM, and DT(decision tree) classifiers are 3.04E-04, 3.00E-03, and 1.03E-01, respectively. Similarly, for the F_{mac}, the p-values for KNN, SVM, and DT classifiers are 3.84E-04, 2.91E-02, and 1.69E-02, respectively. These p-values indicate significant differences exist in both accuracy and F_{mac} among these competing algorithms. Therefore, Fig.5 illustrates the statistical analysis conducted to evaluate the prediction accuracy and F_{mac} of these competing algorithms in the context of KNN, SVM, and DT(decision tree) classifiers.

From Tables 3-8 and Fig.5-6, we can observe:

Table 3: Predictive Accuracy Using KNN as the Classifier

Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.74	0.59	0.69	**0.815**	0.755	0.62	0.555	0.675	0.72	**0.815**
ALLAML	0.8429	0.9286	0.9143	0.9286	0.8714	0.8571	0.9286	0.9	0.9028	**0.9304**
Carcinom	0.684	0.7526	0.684	0.5978	0.6951	**0.8965**	0.6835	0.7408	0.7647	0.862
COIL20	0.9611	0.6056	0.7729	**0.9854**	0.9597	0.9799	0.4583	0.5667	0.8625	0.6944
Colon	0.4833	0.7667	0.7833	0.8	0.75	0.7333	**0.8667**	0.8167	0.7756	0.7923
GLI-85	0.8353	0.7647	**0.8471**	0.8353	0.8118	0.7294	0.8118	0.6706	0.7764	0.8352
GILMO	0.62	0.6	0.62	**0.78**	0.74	0.42	0.4	0.74	0.6799	0.7599
Lung	0.7272	0.6318	0.4523	0.5477	0.641	0.7805	0.4544	0.6472	**0.7942**	0.6857
Lymphomas	0.5263	0.6737	0.5684	0.7263	0.7789	0.8947	0.5474	0.7158	0.8957	**0.9057**
Nci-9	0.1333	0.1667	0.1667	0.2667	0.3667	0.4	0.2	0.1167	0.2666	**0.4833**
Orlraws10P	0.59	0.56	0.52	0.6	0.9	**0.93**	0.67	0.65	0.9	0.9099
Pixraw10P	0.86	0.81	0.66	**0.97**	0.92	0.95	0.63	0.82	0.94	0.95
SMK-CAN-187	0.6108	0.6541	0.6054	0.5676	0.6595	0.5027	0.6162	0.6541	0.6362	**0.6957**
WarpPIE100	0.881	0.5905	0.7524	0.9143	0.8857	**0.9714**	0.8286	0.6286	0.8571	0.8095
AVG.	0.6782	0.6496	0.6455	0.7382	0.7668	0.7618	0.6179	0.6673	0.7694	**0.7949**
AVG. RANKS	6.4643	7.0714	6.9643	4.1429	4.6071	4.8929	7.1071	6.4286	4.4643	**2.8571**

Table 4: Predictive Accuracy Using SVM as the Classifier

Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.7	0.59	0.68	0.76	**0.79**	0.575	0.575	0.705	0.65	0.7299
ALLAML	0.8857	0.9286	0.9286	0.9286	0.8571	0.8286	0.8714	0.9286	0.9438	**0.9714**
Carcinom	0.7067	0.7812	0.6439	0.5916	0.7185	**0.8966**	0.7245	0.8106	0.7932	0.8043
COIL20	0.9896	0.4701	0.6722	0.9931	0.9688	**0.9958**	0.3632	0.4951	0.934	0.7173
Colon	0.7333	0.7167	**0.7833**	0.7333	0.7333	0.7333	0.7167	0.7667	0.7448	0.7435
GLI-85	0.8235	0.7882	0.8118	**0.8588**	0.8235	0.7294	0.8235	0.6706	0.6352	0.7294
GILMO	0.52	0.54	0.6	**0.76**	0.62	0.42	0.36	0.66	0.64	0.72
Lung	0.7026	0.5477	0.4831	0.5118	0.5764	**0.84**	0.4718	0.5918	0.8085	0.7266
Lymphomas	0.4737	0.6211	0.6105	0.7053	0.8	**0.8632**	0.5474	0.7053	0.7926	0.8031
Nci-9	0.1	0.1833	0.1167	0.2667	**0.3833**	0.3333	0.2333	0.0667	0.25	0.2833
Orlraws10P	0.73	0.68	0.47	0.52	0.88	0.95	0.68	0.77	**0.97**	**0.97**
Pixraw10P	0.92	0.92	0.73	0.95	0.96	0.89	0.76	0.84	0.9199	**0.97**
SMK-CAN-187	0.6486	0.6486	0.5784	0.5892	0.6378	0.6108	**0.6595**	0.6432	0.6469	0.6524
WarpPIE100	0.9571	0.5524	0.7	0.9571	0.9333	**0.9762**	0.9048	0.6714	0.9047	0.9095
AVG.	0.7065	0.6406	0.6292	0.7233	0.7630	0.7602	0.6208	0.6661	0.7595	**0.7665**
AVG. RANKS	5.7143	6.8214	7.25	4.8571	4.4643	4.8214	7.4643	5.7857	4.6786	**3.1429**

- UHGSFS *vs.* Alpha-investing: According to the average ranks of statistical tests, UHGSFS outperforms Alpha-investing regarding prediction accuracy and F_{mac} with KNN, SVM, and DT classifiers. Alpha-investing tends to select fewer features than UHGSFS, except for the COIL-20 dataset. Alpha-investing selects only the first few features for some datasets, resulting in the poor prediction accuracy among all competing algorithms. This indicates that Alpha-investing may not be suitable for handling diverse datasets. In contrast, UHGSFS addresses this issue by selecting an appropriate number of features based on the feature clus-

Table 5: Predictive Accuracy Using DT as the Classifier

Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.69	0.515	0.66	**0.735**	0.715	0.63	0.525	0.64	0.68	0.725
ALLAML	0.7571	0.8857	**0.9286**	**0.9286**	0.8714	0.8571	0.9	0.8429	0.8733	0.8342
Carcinom	0.5339	0.575	0.5061	0.5746	**0.6556**	0.5464	0.5002	0.5516	0.6097	0.6379
COIL20	0.8931	0.5993	0.734	**0.8993**	0.8736	0.8799	0.4299	0.5375	0.7493	0.7201
Colon	0.5833	0.75	0.7167	0.75	0.5667	0.6667	**0.7833**	0.7667	0.7282	0.7089
GLI-85	0.7647	0.6353	0.8	**0.8471**	0.7765	0.7529	0.7529	0.6235	0.7294	0.7294
GILMO	0.54	0.4	0.56	**0.58**	0.52	0.46	0.42	0.56	0.4599	0.5599
Lung	0.519	0.4256	0.4256	0.4544	0.3723	0.4523	0.4985	0.4564	0.4666	**0.5628**
Lymphomas	0.3684	0.5158	0.5684	0.5684	0.5053	0.5684	0.4632	**0.5895**	0.5647	0.5742
Nci-9	0.05	0.1667	0.1167	0.2833	0.2833	0.2667	0.1833	0.1	0.3166	**0.3666**
Orlraws10P	0.73	0.67	0.5	0.54	0.74	0.57	0.66	0.76	0.73	**0.8**
Pixraw10P	0.95	0.91	0.68	**0.96**	0.93	0.58	0.78	0.85	0.9199	0.9099
SMK-CAN-187	0.5514	0.5946	0.573	0.5568	0.5892	0.5784	**0.6324**	0.5784	0.5513	0.6044
WarpPIE100	0.7429	0.5762	0.6714	0.7619	**0.7857**	0.6952	0.6571	0.519	0.6238	0.7333
AVG.	0.6196	0.5871	0.6029	0.6742	0.6560	0.6074	0.5847	0.5983	0.6431	**0.6762**
AVG. RANKS	5.8929	6.7143	6.1071	**3.3929**	4.6786	6.4286	6.3214	6	5.5	3.9643

Table 6: F_{mac} Using KNN as the Classifier

Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.7386	0.58	0.6867	**0.8141**	0.7543	0.6078	0.5331	0.668	0.7154	0.8129
ALLAML	0.76	0.8986	0.8951	0.9132	0.8303	0.8367	0.9228	0.8794	0.8852	**0.9252**
Carcinom	0.5768	0.6528	0.5848	0.4909	0.6222	**0.8481**	0.5888	0.6246	0.6678	0.8059
COIL20	0.9593	0.5869	0.7628	**0.9849**	0.9591	0.9815	0.429	0.5508	0.8522	0.6768
Colon	0.4134	0.7252	0.7543	0.7656	0.7315	0.7177	**0.8512**	0.7728	0.7081	0.7493
GLI-85	0.7878	0.726	0.7975	0.7551	0.788	0.6809	0.7421	0.5239	0.7283	**0.8045**
GILMO	0.5228	0.4985	0.5614	**0.7222**	0.6993	0.3829	0.3326	0.7074	0.5817	0.7159
Lung	0.6624	0.5365	0.2799	0.4599	0.5422	**0.7945**	0.279	0.5186	0.7566	0.5705
Lymphomas	0.3306	0.4522	0.3255	0.5567	0.6418	**0.838**	0.2928	0.5774	0.7707	0.7925
Nci-9	0.111	0.1426	0.1787	0.293	0.4022	**0.409**	0.2063	0.1049	0.1992	0.3638
Orlraws10P	0.5884	0.5205	0.4622	0.5738	0.8778	**0.9353**	0.6083	0.6435	0.8853	0.9
Pixraw10P	0.8237	0.8166	0.6141	0.9335	0.922	**0.9597**	0.5952	0.7811	0.9306	0.938
SMK-CAN-187	0.6053	0.6459	0.5969	0.5629	0.6535	0.4435	0.6101	0.6464	0.6191	**0.6805**
WarpPIE100	0.8495	0.5686	0.6921	0.9089	0.8681	**0.9724**	0.835	0.631	0.8497	0.8008
AVG.	0.6235	0.5965	0.5851	0.6953	0.7352	0.7434	0.5590	0.6164	0.7250	**0.7526**
AVG. RANKS	6.5714	7.0714	6.9286	4.5714	4.2857	4.3571	7.0714	6.4286	4.7857	**2.9286**

tering of the target dataset. This strategy ensures performance stability and enhances the UHGSFS's ability to handle various datasets.

- UHGSFS *vs.* SAOLA: Based on the Nmenyi test, UHGSFS significantly performs better than SAOLA in terms of ACC-KNN, F_{mac}-KNN, and F_{mac}-DT. On the other classifiers, UHGSFS achieves lower average ranks than SAOLA. SAOLA employs an innovative online pairwise comparison technique that exclusively considers the relationship between two features. Meanwhile, SAOLA exhibits a lower average number of selected features compared to

Table 7: F_{mac} Using SVM as the Classifier

Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.6978	0.5799	0.6762	0.7582	**0.7873**	0.5665	0.4909	0.6992	0.6426	0.7271
ALLAML	0.8025	0.8962	0.9101	0.9132	0.8265	0.8056	0.8363	0.9151	0.9342	**0.9672**
Carcinom	0.6604	0.6956	0.5415	0.4684	0.6424	**0.8761**	0.6438	0.768	0.6889	0.6625
COIL20	0.9888	0.4291	0.661	0.9936	0.9681	**0.995**	0.3241	0.4372	0.9282	0.6923
Colon	0.497	0.6796	**0.7543**	0.6942	0.7034	0.7152	0.6661	0.717	0.6317	0.6692
GLI-85	0.7685	0.7327	0.7639	**0.8007**	0.7756	0.6673	0.7651	0.5049	0.5976	0.5364
GILMO	0.4633	0.4819	0.5047	**0.6606**	0.5672	0.3763	0.2912	0.6123	0.4963	0.612
Lung	0.6894	0.4377	0.3314	0.4173	0.515	**0.8053**	0.2567	0.5144	0.6854	0.5604
Lymphomas	0.3725	0.4356	0.4306	0.5639	0.6939	**0.7782**	0.393	0.5706	0.5373	0.5467
Nci-9	0.0948	0.1691	0.1244	0.3003	**0.4113**	0.4089	0.262	0.0468	0.1514	0.2064
Orlraws10P	0.7324	0.6392	0.4245	0.4893	0.8639	0.9589	0.6705	0.7745	**0.9686**	0.968
Pixraw10P	0.9277	0.9291	0.6963	**0.9672**	0.9571	0.8962	0.7283	0.8017	0.9039	0.9647
SMK-CAN-187	0.6442	0.6448	0.5738	0.5823	0.6327	0.6079	**0.6559**	0.634	0.6029	0.63491
WarpPIE100	0.9495	0.5331	0.6664	0.9533	0.9159	**0.9743**	0.9044	0.6674	0.8997	0.9069
AVG.	0.6635	0.5917	0.5757	0.6830	0.7329	**0.7451**	0.5635	0.6188	0.6906	0.6896
AVG. RANKS	5.9286	6.3571	7.2857	4.3571	**4.0000**	4.3571	7.2857	5.4286	5.6429	4.3571

Table 8: F_{mac} Using DT as the Classifier

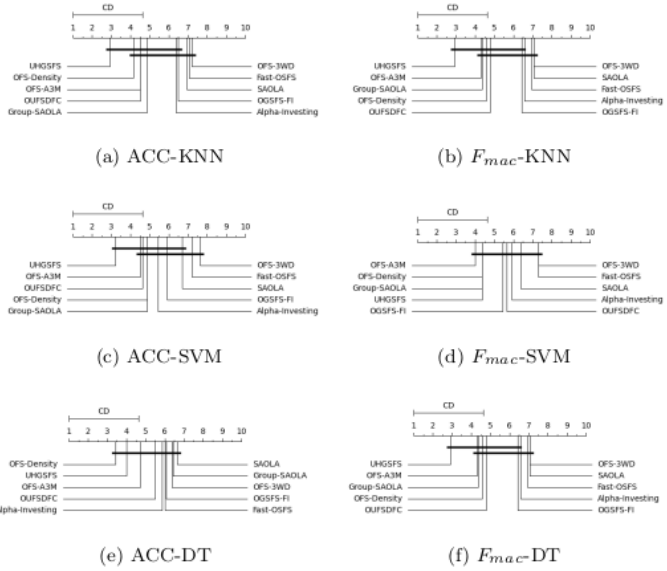
Data Set	Alpha-investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	0.6846	0.501	0.6586	**0.7318**	0.7091	0.6216	0.5134	0.6343	0.6772	0.7184
ALLAML	0.7302	0.8684	0.9195	**0.9207**	0.8479	0.8367	0.8816	0.8152	0.8608	0.8189
Carcinom	0.4485	0.5441	0.4137	0.4702	**0.5876**	0.4682	0.3985	0.4743	0.5005	0.4873
COIL20	0.8903	0.5904	0.726	**0.8965**	0.8722	0.8779	0.4139	0.5289	0.7439	0.7122
Colon	0.4601	0.6936	0.6609	0.7213	0.5147	0.6347	**0.7452**	0.7131	0.6316	0.6821
GLI-85	0.679	0.5467	0.699	**0.7868**	0.7001	0.6896	0.6873	0.5423	0.5536	0.6518
GILMO	0.4579	0.3227	0.4211	0.5133	0.4552	0.409	0.3386	**0.5658**	0.4276	0.4901
Lung	**0.5063**	0.3152	0.2662	0.375	0.3125	0.3579	0.3146	0.3946	0.3894	0.4626
Lymphomas	0.1315	0.2994	0.2872	0.4065	0.391	**0.455**	0.2486	0.3967	0.3871	0.3877
Nci-9	0.0452	0.2106	0.1317	0.3071	0.2676	0.2399	0.1747	0.0758	0.2621	**0.3379**
Orlraws10P	0.7363	0.6319	0.4555	0.5244	0.7525	0.5729	0.656	0.7425	0.702	**0.784**
Pixraw10P	**0.9553**	0.9171	0.618	0.9479	0.9449	0.5492	0.7491	0.8197	0.9146	0.9053
SMK-CAN-187	0.548	0.5932	0.5696	0.5498	0.5819	0.5755	**0.6229**	0.5742	0.5382	0.5924
WarpPIE100	0.7292	0.5603	0.6471	0.7151	**0.785**	0.6824	0.6507	0.4899	0.6183	0.7336
AVG.	0.5716	0.5425	0.5339	**0.6333**	0.6230	0.5693	0.5282	0.5548	0.5862	0.6260
AVG. RANKS	5.8571	6.3571	7	**3.2857**	4	6.0000	6.5714	6.0714	5.8571	4

UHGSFS. However, it is worth noting that SAOLA can not handle mixed type of streaming features.

- UHGSFS *vs.* Fast-OSFS: Based on the Nmenyi test, UHGSFS performs significantly better than SAOLA regarding ACC-KNN, F_{mac}-KNN, ACC-SVM, F_{mac}-DT. Judging from the average number of selected features, Fast-OSFS selects much fewer features than UHGSFS. However, it is worth noting that among all competing algorithms, Fast-OSFS selects the fewest features, which may lead to the loss of important information and thus reduce predic-

Table 9: The mean number of selected features

Data Set	Alpha-Investing	SAOLA	Fast-OSFS	OFS-Density	OFS-A3M	Group-SAOLA	OFS-3WD	OGSFS-FI	OUFSDFC	UHGSFS
Arcene	9.2	26.4	5.2	37.2	31.4	8	3.4	6.2	61	535.8
ALLAML	11.6	30.8	4.4	3.6	12.6	1	11.8	20	120	252.8
Carcinom	30.8	47.4	8.6	5.8	35.4	1337.6	19.6	72.8	91.4	398.6
COIL20	193.4	3.4	7	119.6	17.2	74.4	13.2	3.2	172.2	16.2
Colon	1.8	5.4	2.2	5.6	18.4	5.4	21.8	15	236.8	77.4
GLI-85	16	33.6	5.2	7	14.8	1	12.4	5	227.6	377.2
GILMO	6.2	15	3.4	8.6	21	2.4	2	16.2	144.2	118.6
Lung	13.8	8.2	3	4	8.6	93.8	2	10.6	99.8	14.6
Lymphomas	9.8	21.4	4.6	7.4	15	715.2	6.4	32.2	241.6	319.6
Nci-9	5.4	17.6	3.6	7.2	24	215	26.8	10.2	274.4	287.6
Orlraws10P	9.8	5.6	3.2	2.6	13.2	279.6	25.2	4.8	776.0	438.4
Pixraw10P	10.4	7.6	4.2	166.2	7.4	287	20.6	2.6	96.4	359.2
SMK-CAN-187	14.8	11.2	5	15	31.6	1	26.4	28.2	96.6	607.8
WarpPIE100	45	3.6	4.6	28.8	29.4	27	24.2	6.2	139.4	94.0
AVG.	27.0	16.9	4.6	29.9	20.0	217.7	15.4	16.7	198.4	278.4
AVG. RANKS	5.00	4.75	2.14	4.43	6.07	5.61	4.29	4.57	9.07	9.07



(a) ACC-KNN (b) F_{mac}-KNN

(c) ACC-SVM (d) F_{mac}-SVM

(e) ACC-DT (f) F_{mac}-DT

Figure 5: The statistical test graph of these competing algorithms

tion accuracy.

- UHGSFS *vs*. OFS-Density: UHGSFS and OFS-Density exhibit comparable prediction accuracy. Specifically, on average predictive accuracy and F_{mac}, UHGSFS demonstrates superior

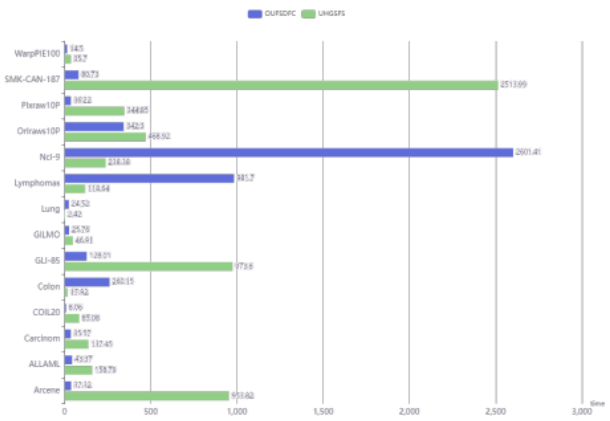


Figure 6: Running time (seconds).

to OFS-Density in most cases of KNN, SVM, and DT algorithms. Regarding the number of selected features, OFS-Density selects more features than UHGSFS on the COIL20 dataset. In contrast, OFS-Density selects significantly fewer features than UHGSFS for other datasets. OFS-Density is a neighborhood rough set-based method, which exhibits high time complexity for large sample datasets. Furthermore, OFS-Density utilizes neighborhood density information for feature selection and may not handle unevenly distributed datasets effectively.

- UHGSFS *vs*. OFS-A3M: There is no significant difference between UHGSFS and OFS-A3M on prediction accuracy. On average predictive accuracy and F_{mac}, UHGSFS performs better than OFS-A3M in the cases of KNN, SVM, and DT. Like OFS-Density, OFS-A3M is a neighborhood rough set-based method with high time complexity for large sample datasets. Regarding the number of selected features, OFS-A3M selects significantly fewer features than UHGSFS. OFS-A3M utilizes neighborhood information for feature selection, which is greatly influenced by the distribution of samples.

- UHGSFS *vs*. OFS-3WD: Based on the Nmenyi test, UHGSFS outperforms significantly better than OFS-3WD in terms of accuracy for KNN and SVM algorithms. Meanwhile, UHGSFS demonstrates significantly superior performance over OFS-3WD in terms of F_{mac} for KNN and DT algorithms. Regarding the average number of selected features, OFS-3WD selects fewer features than UHGSFS. OFS-3WD utilizes global dynamics and three-way decision-making

techniques for feature selection. However, it is essential to note that OFS-3WD cannot handle heterogeneous streaming features, limiting its applicability in such scenarios.

- UHGSFS *vs*. Group-SAOLA: There is no statistically significant difference in prediction accuracy and F_{mac} between UHGSFS and Group-SAOLA. However, on average, UHGSFS outperforms Group-SAOLA in most cases. Regarding the average number of selected features, Group-SAOLA selects much more features than UHGSFS on datasets Carcinom, COIL20, and Lymphomas, while it selects much fewer features on the remaining datasets. Group-SAOLA is an online streaming group feature selection method that exhibits fast execution on ultra-high-dimensional datasets. However, it does not consider feature relationships between streaming groups. Group-SAOLA selects a significantly smaller number of features (just one feature) on some datasets, potentially losing important information.

- UHGSFS *vs*. OGSFS-FI: Regarding accuracy (ACC) and F_{mac}, UHGSFS exhibits higher average prediction accuracy and lower average ranks than OGSFS-FI for KNN, SVM, and DT classifiers. Additionally, OGSFS-FI selects fewer features on average than UHGSFS. OGSFS-FI is an online group streaming feature selection method that employs mutual information for feature selection, but it cannot handle heterogeneous streaming features.

- UHGSFS *vs*. OUFSDFC: Both UHGSFS and OUFSDFC are unsupervised online streaming feature selection algorithms. Regarding accuracy (ACC) and F_{mac}, UHGSFS demonstrates higher average prediction accuracy and lower average ranks than OUFSDFC. On the average number of selected features, UHGSFS and OUFSDFC select about the same. Analyzing the running times depicted in Fig.6, it is observed that UHGSFS exhibits shorter running times for discrete datasets (e.g., lung, NCI-9) compared to OUFSDFC. Conversely, for continuous datasets (e.g., COIL20), OUFSDFC requires less time than UHGSFS. OUFSDFC requires more time to discover the relationship between two features when processing continuous streaming features. It is important to note that OUFSDFC employs separate metrics for processing continuous and discrete datasets, making it incapable of handling scenarios where the feature streaming type is unknown.

In sum, UHGSFS performs excellently on average accuracy (ACC) and F_{mac} in cases of KNN, SVM, and DT classifiers. Even without label information, UHGSFS exhibits statistically better or

equivalent performance compared to supervised streaming feature selection methods. Furthermore, compared to competing algorithms, UHGSFS indicates enhanced scalability regarding running time and the number of selected features. Importantly, UHGSFS can effectively handle heterogeneous streaming features without knowing the information about feature types in advance.

4.3. The Effectiveness of UHGSFS in Handling Heterogeneous Streaming Features

Since the data sets in Table 2 are continuous or discrete feature datasets. To verify the effectiveness of our UHGSFS algorithm in handling heterogeneous streaming features, we take the continuous datasets in Table 2, randomly select 50% features, and discretize these features into ten equal parts. Then, we conduct experiments on these two types of datasets (original and mixed). The experimental results on prediction accuracy and F_{mac} are shown in Tables 10 and 11, where KNN-ACC-MIX, SVM-ACC-MIX, and DT-ACC-MIX denote the prediction accuracy of UHGSFS on mixed streaming features.

Table 10: ACC of Heterogeneous Streaming Features

Data Set	KNN-ACC	KNN-ACC-MIX	SVM-ACC	SVM-ACC-MIX	DT-ACC	DT-ACC-MIX
Arcene	0.8150	0.8350	0.7299	0.7250	0.7250	0.7750
ALLAML	0.9304	0.9314	0.9714	0.9038	0.8342	0.9171
Carcinom	0.8620	0.8213	0.8043	0.8042	0.6379	0.5916
COIL20	0.6944	0.7840	0.7173	0.8146	0.7201	0.7576
GLI-85	0.8352	0.7882	0.7294	0.7529	0.7294	0.8118
GILMO	0.7599	0.7400	0.7200	0.6600	0.5599	0.6000
Orlraws10P	0.9099	0.9100	0.9700	0.9600	0.8000	0.7700
Pixraw10P	0.9500	0.9300	0.9700	0.9700	0.9099	0.8800
SMK-CAN-187	0.6957	0.6209	0.6524	0.6633	0.6044	0.5192
WarpPIE100	0.8095	0.8286	0.9095	0.8619	0.7333	0.7000
AVG.	0.8262	0.8190	0.8174	0.8116	0.7254	0.7322

From Tables 10 and 11, we can observe that for some datasets, the selected features on mixed version and the final classification ACC and F_{mac} do not perform as well as when using the original dataset. Meanwhile, features selected on the mixed dataset outperform the original dataset on some other datasets. This fluctuation is slight, and the root cause is the instability of the feature selection algorithm. Overall, our new algorithm performs essentially equally well on both two kinds of datasets. These results demonstrate our method's capability when dealing with heterogeneous streaming features.

Table 11: F_{mac} of Heterogeneous Streaming Features

Data Set	KNN-F_{mac}	KNN-F_{mac}-MIX	SVM-F_{mac}	SVM-F_{mac}-MIX	DT-F_{mac}	DT-F_{mac}-MIX
Arcene	0.8129	0.8321	0.7271	0.7221	0.7184	0.7708
ALLAML	0.9252	0.9166	0.9672	0.8881	0.8189	0.9047
Carcinom	0.8059	0.8213	0.6625	0.6848	0.4873	0.4859
COIL20	0.6768	0.7709	0.6923	0.8055	0.7122	0.7513
GLI-85	0.8045	0.7504	0.5364	0.5795	0.6518	0.7777
GILMO	0.7159	0.7033	0.6120	0.5060	0.4901	0.5535
Orlraws10P	0.9000	0.9060	0.9680	0.9533	0.7840	0.7567
Pixraw10P	0.9380	0.9168	0.9647	0.9627	0.9053	0.8707
SMK-CAN-187	0.6805	0.5922	0.6349	0.6433	0.5924	0.4994
WarpPIE100	0.8008	0.8125	0.9069	0.8592	0.7336	0.6900
AVG.	0.8061	0.8022	0.7672	0.7604	0.6894	0.7061

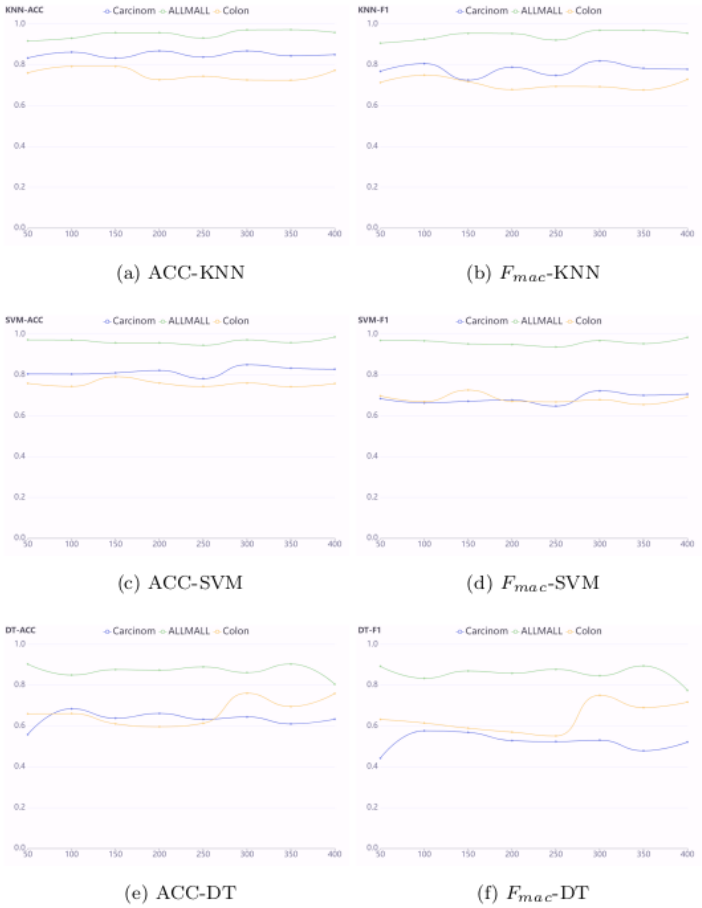
4.4. Analysis of Streaming Group Size

In order to investigate the effect of different group sizes on the UHGSFS, we chose three representative datasets, Carcinom, ALLAML, and Colon, for parameter analysis. Carcinom and ALLAML are continuous datasets, while Colon is a discrete dataset. We set the group size to increase from 50 to 400, and the interval between each increase was 50. Fig.7 shows the accuracy and F_{mac} curves for KNN, SVM, and DT classifiers varying with different group sizes, respectively.

By observing the experimental results, we can find that ACC and F_{mac} performance on these three datasets presents little change regardless of different group sizes regarding KNN, SVM, and DT classifiers. This indicates that group size does not significantly affect the results of UHGSFS.

5. Conclusion

In this paper, we have developed an unsupervised online streaming feature selection method to handle heterogeneous unknown-type streaming features. Specifically, we employ MIC to calculate the relationship between heterogeneous streaming features. Then, we design an adaptive feature clustering algorithm to reduce redundancy among features. Representative features from each feature cluster are selected by maximizing feature relevance and minimizing redundancy. Experimental results and comparative studies demonstrate that, without label information, the UHGSFS method statistically outperforms or performs equally well as state-of-the-art supervised streaming feature selection methods. However, due to the computational demands of MIC, UHGSFS exhibits higher



(a) ACC-KNN (b) F_{mac}-KNN

(c) ACC-SVM (d) F_{mac}-SVM

(e) ACC-DT (f) F_{mac}-DT

Figure 7: Accuracy and F_{mac} curves of UHGSFS varying in different sizes of the group.

time complexity. In the future, we will focus on distributed methods to address the time complexity issue of UHGSFS.

Acknowledgment

This work is supported in part by the National Natural Science Foundation of China under grants 62376001, the Natural Science Foundation of Anhui Province of China under grants 2308085MF215.

References

[1] H. Liu, H. Motoda, Computational Methods of Feature Selection, Chapman and Hall/CRC Press, 2007.

[2] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, H. Liu, Feature selection: A data perspective, ACM Computing Surveys 50 (6) (2018) 1–45.

[3] S. Solorio-Fernández, J. A. Carrasco-Ochoa, J. F. Martínez-Trinidad, A review of unsupervised feature selection methods, Artificial Intelligence Review 53 (2) (2020) 907–948.

[4] E. Hancer, B. Xue, M. Zhang, A survey on feature selection approaches for clustering, Artificial Intelligence Review 53 (2020) 4519–4545.

[5] J. Cai, J. Luo, S. Wang, S. Yang, Feature selection in machine learning: A new perspective, Neurocomputing 300 (2018) 70–79.

[6] M. Wang, H. Li, D. Tao, K. Lu, X. Wu, Multimodal graph-based reranking for web image search, IEEE transactions on image processing 21 (11) (2012) 4649–4661.

[7] W. Ding, T. F. Stepinski, Y. Mu, L. Bandeira, R. Ricardo, Y. Wu, Z. Lu, T. Cao, X. Wu, Subkilometer crater discovery with boosting and transfer learning, ACM Transactions on Intelligent Systems and Technology (TIST) 2 (4) (2011) 1–22.

[8] D. Lei, P. Liang, J. Hu, Y. Yuan, New online streaming feature selection based on neighborhood rough set for medical data, Symmetry 12 (10) (2020) 1635.

[9] X. Wu, K. Yu, W. Ding, H. Wang, X. Zhu, Online feature selection with streaming features, IEEE transactions on pattern analysis and machine intelligence 35 (5) (2012) 1178–1192.

[10] P. Zhou, S. Zhao, Y. Yan, X. Wu, Online scalable streaming feature selection via dynamic decision, ACM Transactions on Knowledge Discovery from Data (TKDD) 16 (5) (2022) 1–20.

[11] P. Zhou, X. Hu, P. Li, X. Wu, Ofs-density: A novel online streaming feature selection method, Pattern Recognition 86 (2019) 48–61.

[12] D. You, Y. Wang, J. Xiao, Y. Lin, M. Pan, Z. Chen, L. Shen, X. Wu, Online multi-label streaming feature selection with label correlation, IEEE Transactions on Knowledge and Data Engineering (2021).

[13] D. Wu, Y. He, X. Luo, M. Zhou, A latent factor analysis-based approach to online sparse streaming feature selection, IEEE Transactions on Systems, Man, and Cybernetics: Systems 52 (11) (2021) 6744–6758.

[14] J. Li, X. Hu, J. Tang, H. Liu, Unsupervised streaming feature selection in social media, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, 2015, pp. 1041–1050.

[15] N. Almusallam, Z. Tari, J. Chan, A. Fahad, A. Alabdulatif, M. Al-Naeem, Towards an unsupervised feature selection method for effective dynamic features, IEEE Access 9 (2021) 77149–77163.

[16] X. Yan, A. Homaifar, M. Sarkar, B. Lartey, K. D. Gupta, An online unsupervised streaming features selection through dynamic feature clustering, IEEE Transactions on Artificial Intelligence (2022).

[17] W. Zheng, S. Chen, Z. Fu, J. Li, J. Yang, Streaming feature selection via graph diffusion, Information Sciences 618 (2022) 150–168.

[18] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, P. C. Sabeti, Detecting novel associations in large data sets, science 334 (6062) (2011) 1518–1524.

[19] Q. Gu, Z. Li, J. Han, Generalized fisher score for feature selection, arXiv preprint arXiv:1202.3725 (2012).

[20] M. Robnik-Šikonja, I. Kononenko, Theoretical and empirical analysis of relieff and rrelieff, Machine learning 53 (2003) 23–69.

[21] J. R. Vergara, P. A. Estévez, A review of feature selection methods based on mutual information, Neural computing and applications 24 (2014) 175–186.

[22] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Advances in neural information processing systems 18 (2005).

[23] P. Mitra, C. Murthy, S. K. Pal, Unsupervised feature selection using feature similarity, IEEE transactions on pattern analysis and machine intelligence 24 (3) (2002) 301–312.

[24] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, in: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, 2010, pp. 333–342.

[25] W. Li, H. Chen, T. Li, J. Wan, B. Sang, Unsupervised feature selection via self-paced learning and low-redundant regularization, Knowledge-Based Systems 240 (2022) 108150.

[26] Y. Du, X. Zhou, C. Yang, T. Huang, An interactive feature selection method based on multi-step state transition algorithm for high-dimensional data, Knowledge-Based Systems (2023) 111102.

[27] Y. Wang, J. Wang, H. Liao, H. Chen, An efficient semi-supervised representatives feature selection algorithm based on information theory, Pattern Recognition 61 (2017) 511–523.

[28] Z. Qiu, W. Zeng, D. Liao, N. Gui, A-sfs: Semi-supervised feature selection based on multi-task self-supervision, Knowledge-Based Systems 252 (2022) 109449.

[29] S. Perkins, J. Theiler, Online feature selection using grafting, in: Proceedings of the 20th International Conference on Machine Learning (ICML-03), 2003, pp. 592–599.

[30] J. Zhou, D. P. Foster, R. A. Stine, L. H. Ungar, I. Guyon, Streamwise feature selection., Journal of Machine Learning Research 7 (9) (2006).

[31] K. Yu, X. Wu, W. Ding, J. Pei, Scalable and accurate online feature selection for big data, ACM Transactions on Knowledge Discovery from Data (TKDD) 11 (2) (2016) 1–39.

[32] D. You, R. Li, S. Liang, M. Sun, X. Ou, F. Yuan, L. Shen, X. Wu, Online causal feature selection for streaming features, IEEE Transactions on Neural Networks and Learning Systems (2021).

[33] P. Zhou, X. Hu, P. Li, X. Wu, Online feature selection for high-dimensional class-imbalanced data, Knowledge-Based Systems 136 (2017) 187–199.

[34] P. Zhou, X. Hu, P. Li, X. Wu, Online streaming feature selection using adapted neighborhood rough set, Information Sciences 481 (2019) 258–279.

[35] J. Liu, Y. Lin, J. Du, H. Zhang, Z. Chen, J. Zhang, Asfs: A novel streaming feature selection for multi-label data based on neighborhood rough set, Applied Intelligence 53 (2) (2023) 1707–1724.

[36] X. Yan, M. Razeghi-Jahromi, A. Homaifar, B. A. Erol, A. Girma, E. Tunstel, A novel streaming data clustering algorithm based on fitness proportionate sharing, IEEE Access 7 (2019) 184985–185000.

[37] D.-X. Chang, X.-D. Zhang, C.-W. Zheng, D.-M. Zhang, A robust dynamic niching genetic algorithm with niche migration for automatic clustering problem, Pattern recognition 43 (4) (2010) 1346–1360.

[38] P. Zhou, N. Wang, S. Zhao, Online group streaming feature selection considering feature interaction, Knowledge-Based Systems 226 (2021) 107157.

[39] X. Hu, P. Zhou, P. Li, J. Wang, X. Wu, A survey on online feature selection with streaming features, Frontiers of Computer Science 12 (2018) 479–493.

[40] D. G. Pereira, A. Afonso, F. M. Medeiros, Overview of friedman's test and post-hoc analysis, Communications in Statistics-Simulation and Computation 44 (10) (2015) 2636–2653.

UHGSFS_1_

ORIGINALITY REPORT

21%
SIMILARITY INDEX

PRIMARY SOURCES

1	Peng Zhou, Yunyun Zhang, Yuanting Yan, Shu Zhao. "Unknown Type Streaming Feature Selection via Maximal Information Coefficient", 2022 IEEE International Conference on Data Mining Workshops (ICDMW), 2022 Crossref	404 words — 4%
2	par.nsf.gov Internet	358 words — 3%
3	Peng Zhou, Shu Zhao, Yuanting Yan, Xindong Wu. "Online Scalable Streaming Feature Selection via Dynamic Decision", ACM Transactions on Knowledge Discovery from Data, 2022 Crossref	145 words — 1%
4	Jingwen Xiong, Wenke Zang, Jing Che, Yuzhen Zhao, Xiyu Liu. "Density Peaks Clustering Based on Natural Search Neighbors and Manifold Distance Metric", IEEE Access, 2022 Crossref	97 words — 1%
5	Peng Zhou, Ni Wang, Shu Zhao. "Online group streaming feature selection considering feature interaction", Knowledge-Based Systems, 2021 Crossref	93 words — 1%
6	coek.info Internet	80 words — 1%

7	arxiv.org Internet	61 words — 1%
8	Shifei Ding, Wei Du, Xiao Xu, Tianhao Shi, Yanru Wang, Chao Li. "An improved density peaks clustering algorithm based on natural neighbor with a merging strategy", Information Sciences, 2023 Crossref	42 words — < 1%
9	Xuyang Yan, Abdollah Homaifar, Mrinmoy Sarkar, Benjamin Lartey, Kishor Datta Gupta. "An Online Unsupervised Streaming Features Selection Through Dynamic Feature Clustering", IEEE Transactions on Artificial Intelligence, 2022 Crossref	39 words — < 1%
10	Xuyang Yan, Abdollah Homaifar, Mrinmoy Sarkar, Benjamin Lartey, Kishor Datta Gupta. "An Online Unsupervised Streaming Features Selection Through Dynamic Feature Clustering", IEEE Transactions on Artificial Intelligence, 2023 Crossref	35 words — < 1%
11	www.arxiv-vanity.com Internet	33 words — < 1%
12	Ezzatul Akmal Kamaru Zaman, Azlinah Mohamed, Azlin Ahmad. "Feature selection for online streaming high-dimensional data: A state-of-the-art review", Applied Soft Computing, 2022 Crossref	27 words — < 1%
13	Peng Zhou, Peipei Li, Shu Zhao, Yanping Zhang. "Online early terminated streaming feature selection based on Rough Set theory", Applied Soft Computing, 2021 Crossref	26 words — < 1%

14	Shengxing Bai, Yaojin Lin, Yan Lv, Jinkun Chen, Chenxi Wang. "Kernelized fuzzy rough sets based online streaming feature selection for large-scale hierarchical classification", Applied Intelligence, 2020 Crossref	24 words — < 1%
15	"Rough Sets and Knowledge Technology", Springer Science and Business Media LLC, 2013 Crossref	23 words — < 1%
16	Peng Zhou, Peipei Li, Shu Zhao, Xindong Wu. "Feature Interaction for Streaming Feature Selection", IEEE Transactions on Neural Networks and Learning Systems, 2020 Crossref	23 words — < 1%
17	"Database Systems for Advanced Applications", Springer Science and Business Media LLC, 2020 Crossref	21 words — < 1%
18	Shixuan Zhou, Peng Song, Yanwei Yu, Wenming Zheng. "Structural regularization based discriminative multi-view unsupervised feature selection", Knowledge-Based Systems, 2023 Crossref	21 words — < 1%
19	export.arxiv.org Internet	20 words — < 1%
20	www.researchgate.net Internet	20 words — < 1%
21	link.springer.com Internet	19 words — < 1%
22	researcharchive.vuw.ac.nz Internet	19 words — < 1%

23	static.tongtianta.site Internet	19 words — < 1%
24	Peng Zhou, Yunyun Zhang, Peipei Li, Xindong Wu. "General assembly framework for online streaming feature selection via Rough Set models", Expert Systems with Applications, 2022 Crossref	18 words — < 1%
25	ebin.pub Internet	18 words — < 1%
26	Yizhang Zou, Xuegang Hu, Peipei Li, Junlong Li. "Multi-Label Streaming Feature Selection via Class-Imbalance Aware Rough Set", 2021 International Joint Conference on Neural Networks (IJCNN), 2021 Crossref	17 words — < 1%
27	Jinghua Liu, Wei Wei, Yaojin Lin, Lijie Yang, Hongbo Zhang. "Learning implicit labeling-importance and label correlation for multi-label feature selection with streaming labels", Pattern Recognition, 2024 Crossref	16 words — < 1%
28	Mengshi Huang, Hongmei Chen, Yong Mi, Chuan Luo, Shi-Jinn Horng, Tianrui Li. "Joint sparse latent representation learning and dual manifold regularization for unsupervised feature selection", Knowledge-Based Systems, 2023 Crossref	16 words — < 1%
29	mafiadoc.com Internet	15 words — < 1%
30	www.scitepress.org Internet	14 words — < 1%

31	Gaoteng Yuan, Yi Zhai, Jiansong Tang, Xiaofeng Zhou. "CSCIM_FS: Cosine similarity coefficient and information measurement criterion-based feature selection method for high-dimensional data", Neurocomputing, 2023 Crossref	13 words — < 1%
32	Peng Zhou, Xuegang Hu, Peipei Li, Xindong Wu. "OFS-Density: A Novel Online Streaming Feature Selection Method", Pattern Recognition, 2018 Crossref	13 words — < 1%
33	Tao Lei, Hulin Liu, Yong Wan, Chenxia Li, Yong Xia, Asoke K. Nandi. "Shape-Guided Dual Consistency Semi-Supervised Learning Framework for 3-D Medical Image Segmentation", IEEE Transactions on Radiation and Plasma Medical Sciences, 2023 Crossref	13 words — < 1%
34	Yan Lv, Yaojin Lin, Xiangyan Chen, Chenxi Wang, Shaozi Li. "Feature interaction based online streaming feature selection via buffer mechanism", Concurrency and Computation: Practice and Experience, 2021 Crossref	13 words — < 1%
35	Zhaolong Ling, Bo Li, Yiwen Zhang, Ying Li, Haifeng Ling. "Online Markov Blanket Learning for High-Dimensional Data", Applied Intelligence, 2022 Crossref	13 words — < 1%
36	selab.iecs.fcu.edu.tw Internet	13 words — < 1%
37	www.ideals.illinois.edu Internet	13 words — < 1%

38	Jingwen Xiong, Wenke Zang, Jing Che, Yuzhen Zhao, Xiyu Liu. "Density Peaks Clustering based on Natural Search Neighbors and Manifold Distance Metric", IEEE Access, 2022 Crossref	12 words —	< 1%
39	Pengfei Zhu, Wencheng Zhu, Qinghua Hu, Changqing Zhang, Wangmeng Zuo. "Subspace clustering guided unsupervised feature selection", Pattern Recognition, 2017 Crossref	12 words —	< 1%
40	Dipti Theng, Kishor K. Bhoyar. "Feature selection techniques for machine learning: a survey of more than two decades of research", Knowledge and Information Systems, 2023 Crossref	11 words —	< 1%
41	Hoseini, Elham, and Eghbal G. Mansoori. "Selecting discriminative features in social media data: An unsupervised approach", Neurocomputing, 2016. Crossref	11 words —	< 1%
42	Vidya Sudevan, Nikita Mankovskii, Sajid Javed, Hamad Karki, Giulia De Masi, Jorge Dias. "Multisensor fusion for marine infrastructures' inspection and safety", OCEANS 2022, Hampton Roads, 2022 Crossref	11 words —	< 1%
43	scholarworks.uaeu.ac.ae Internet	11 words —	< 1%
44	www.mdpi.com Internet	11 words —	< 1%
45	D. Wang, M. Hollaus, N. Pfeifer. "FEASIBILITY OF MACHINE LEARNING METHODS FOR	10 words —	< 1%

SEPARATING WOOD AND LEAF POINTS FROM TERRESTRIAL LASER SCANNING DATA", ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, 2017
Crossref

46 JianGang Lu, An Zhou, Qinqin Wu, Xiaoyan Liu, Yiyuan Bao. "Method of Feature Selection via the Feature Ensemble Clustering and Selecting", 2021 China Automation Congress (CAC), 2021 — 10 words — < 1%
Crossref

47 Noura Al Nuaimi, Mohammad M. Masud. "Online streaming feature selection with incremental feature grouping", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2020 — 10 words — < 1%
Crossref

48 Peng Zhou, Xuegang Hu, Peipei Li, Xindong Wu. "Online feature selection for high-dimensional class-imbalanced data", Knowledge-Based Systems, 2017 — 10 words — < 1%
Crossref

49 Xianchun Zhou, Mengjia Fan. "Four-Directional Total Variation With Overlapping Group Sparsity for Image Denosing", IEEE Access, 2021 — 10 words — < 1%
Crossref

50 core.ac.uk — 10 words — < 1%
Internet

51 www.degruyter.com — 10 words — < 1%
Internet

52 "PRICAI 2019: Trends in Artificial Intelligence", Springer Science and Business Media LLC, 2019 — 9 words — < 1%
Crossref

53 Azar Rafie, Parham Moradi, Abdulbaghi Ghaderzadeh. "A Multi-Objective Online Streaming Multi-Label Feature Selection using Mutual information", Expert Systems with Applications, 2022
Crossref
9 words — < 1%

54 Bolón-Canedo, V., N. Sánchez-Maroño, and A. Alonso-Betanzos. "Recent advances and emerging challenges of feature selection in the context of big data", Knowledge-Based Systems, 2015.
Crossref
9 words — < 1%

55 Lecture Notes in Computer Science, 2011.
Crossref
9 words — < 1%

56 Mohamed Aly Bouke, Azizol Abdullah, Jaroslav Frnda, Korhan Cengiz, Bashir Salah. "Bukagini: a Stability-Aware Gini Index Feature Selection Algorithm for Robust Model Performance", IEEE Access, 2023
Crossref
9 words — < 1%

57 deepai.org
Internet
9 words — < 1%

58 Chao Sheng, Peng Song, Weijian Zhang, Dongliang Chen. "Dual-graph regularized subspace learning based feature selection", Digital Signal Processing, 2021
Crossref
8 words — < 1%

59 Dianlong You, Miaomiao Sun, Shunpan Liang, Ruiqi Li, Yang Wang, Jiawei Xiao, Fuyong Yuan, Limin Shen, Xindong Wu. "Online Feature Selection for Multi-Source Streaming Features", Information Sciences, 2022
Crossref
8 words — < 1%

60 Dipanjyoti Paul, Rahul Kumar, Sriparna Saha, Jimson Mathew. "Multi-objective Cuckoo Search-
8 words — < 1%

based Streaming Feature Selection for Multi-label Dataset", ACM Transactions on Knowledge Discovery from Data, 2021
Crossref

61 Dongxia Chang, Yao Zhao, Lian Liu, Changwen Zheng. "A dynamic niching clustering algorithm based on individual-connectedness and its application to color image segmentation", Pattern Recognition, 2016 — 8 words — < 1%
Crossref

62 Haishuai Wang, Peng Zhang, Xingquan Zhu, Ivor Wai-Hung Tsang, Ling Chen, Chengqi Zhang, Xindong Wu. "Incremental Subgraph Feature Selection for Graph Classification", IEEE Transactions on Knowledge and Data Engineering, 2017 — 8 words — < 1%
Crossref

63 Lijun Zhang, , Chun Chen, Jiajun Bu, and Xiaofei He. "A Unified Feature and Instance Selection Framework Using Optimum Experimental Design", IEEE Transactions on Image Processing, 2012. — 8 words — < 1%
Crossref

64 Mahsa Samareh-Jahani, Farid Saberi-Movahed, Mahdi Eftekhari, Gholamreza Aghamollaei, Prayag Tiwari. "Low-redundant Unsupervised Feature Selection based on Data Structure Learning and Feature Orthogonalization", Expert Systems with Applications, 2023 — 8 words — < 1%
Crossref

65 Muhammed Abd-Elnaby, Marco Alfonse, Mohamed Roushdy. "Classification of breast cancer using microarray gene expression data: A survey", Journal of Biomedical Informatics, 2021 — 8 words — < 1%
Crossref

66 Pengfei Zhu, Wangmeng Zuo, Lei Zhang, Qinghua Hu, Simon C.K. Shiu. "Unsupervised feature — 8 words — < 1%

selection by regularized self-representation", Pattern Recognition, 2015
Crossref

67 Wenhao Shu, Jianhui Yu, Zhenchao Yan, Wenbin Qian. "Semi-supervised feature selection for partially labeled mixed-type data based on multi-criteria measure approach", International Journal of Approximate Reasoning, 2022 — 8 words — < 1%
Crossref

68 Yadi Wang, Jun Wang. "Neurodynamics-driven holistic approaches to semi-supervised feature selection", Neural Networks, 2022 — 8 words — < 1%
Crossref

69 Yangyi Du, Xiaojun Zhou, Chunhua Yang, Tingwen Huang. "An interactive feature selection method based on multi-step state transition algorithm for high-dimensional data", Knowledge-Based Systems, 2023 — 8 words — < 1%
Crossref

70 Zhaleh Manbari, Fardin Akhlaghian Tab, Chiman Salavati. "Hybrid Fast Unsupervised Feature Selection for High-dimensional Data", Expert Systems with Applications, 2019 — 8 words — < 1%
Crossref

71 lcs.ios.ac.cn — 8 words — < 1%
Internet

72 library2.usask.ca — 8 words — < 1%
Internet

73 oaji.net — 8 words — < 1%
Internet

repository.asu.edu

74	Internet	8 words — < 1%
75	researcharchive.victoria.ac.nz Internet	8 words — < 1%
76	scholarworks.gsu.edu Internet	8 words — < 1%
77	www.cs.umb.edu Internet	8 words — < 1%
78	Noura AlNuaimi, Mohammad Mehedy Masud, Mohamed Adel Serhani, Nazar Zaki. "Streaming Feature Selection Algorithms for Big Data: A Survey", Applied Computing and Informatics, 2019 Crossref	7 words — < 1%
79	Yu Song, Zhigang Liu. "An Unsupervised Online Streaming Feature Selection Algorithm with Density Peak Clustering", 2023 IEEE International Conference on Networking, Sensing and Control (ICNSC), 2023 Crossref	7 words — < 1%
80	Jie Cai, Jiawei Luo, Shulin Wang, Sheng Yang. "Feature selection in machine learning: A new perspective", Neurocomputing, 2018 Crossref	6 words — < 1%
81	Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, Huan Liu. "Feature Selection", ACM Computing Surveys, 2017 Crossref	6 words — < 1%
82	Wanghui Xiao, Di Wu. "An Improved Siamese Network Model for Handwritten Signature	6 words — < 1%

Verification", 2021 IEEE International Conference on Networking, Sensing and Control (ICNSC), 2021
Crossref

83 Xiao-Ting Wang, Xin-Ze Luan. "Bayesian Penalized Method for Streaming Feature Selection", IEEE Access, 2019 — 6 words — < 1%
Crossref

84 Yan Lv, Yaojin Lin, Xiangyan Chen, Dongxing Wang, Chenxi Wang. "Online Streaming Feature Selection Based on Feature Interaction", 2020 IEEE International Conference on Knowledge Graph (ICKG), 2020 — 6 words — < 1%
Crossref

85 Yintong Wang. "Unsupervised Representative Feature Selection Algorithm Based on Information Entropy and Relevance Analysis", IEEE Access, 2018 — 6 words — < 1%
Crossref

86 Yun Li, Tao Li, Huan Liu. "Recent advances in feature selection and its applications", Knowledge and Information Systems, 2017 — 6 words — < 1%
Crossref

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF