2022 IEEE International Conference on Data Mining Workshops (ICDMW)

Unknown Type Streaming Feature Selection via Maximal Information Coefficient

Peng Zhou, Yunyun Zhang, Yuanting Yan, Shu Zhao
Key Laboratory of Intelligent Computing and Signal Processing (the Ministry of Education of China)
School of Computer Science and Technology, Anhui University, Hefei, China
doodzhou@ahu.edu.cn, zhangyunyun1110@stu.ahu.edu.cn, ytyan@ahu.edu.cn, zhaoshuzs@ahu.edu.cn

***Abstract*—Feature selection aims to select an optimal minimal feature subset from the original datasets and has become an indispensable preprocessing component before data mining and machine learning, especially in the era of big data. Most feature selection methods implicitly assume that we can know the feature type (categorical, numerical, or mixed) before learning, then design corresponding measurements to calculate the correlation between features. However, in practical applications, features may be generated dynamically and arrive one by one over time, which we call streaming features. Most existing streaming feature selection methods assume that all dynamically generated features are the same type or assume we can know the feature type for each new arriving feature on the fly, but this is unreasonable and unrealistic. Therefore, this paper firstly studies a practical issue of Unknown Type Streaming Feature Selection and proposes a new method to handle it, named UT-SFS. Extensive experimental results indicate the effectiveness of our new method. UT-SFS is nonparametric and does not need to know the feature type before learning, which aligns with practical application needs.**

***Index Terms*—feature selection, streaming feature, unknown feature type, maximal information coefficient**

I. INTRODUCTION

Feature selection aims to select the smallest sized subset of the original feature space that preserves the best salient features required from the dataset [1]. With the explosive growth of data volume and dimension, feature selection has become a necessary data preprocessing technique that is widely used in data mining, machine learning, and other fields [2]. By removing noisy, irrelevant, and redundant features, machine learning can gain significant benefits from feature selection, such as better performance, less running time, and better understandability [3], [4].

Traditional feature selection assumes that the entire feature space can be fully presented to the learner before learning [5]. To select an optimal feature subset, feature selection algorithms tend to traverse the entire dataset multiple times. However, in real-world applications, such as image analysis [6] and Martian crater detection [7], not all features can be acquired before learning. Features can be generated and arrive one by one over time, while the number of samples remains fixed, which we call streaming features [8]. For example, because the high cost of conducting wet-lab experiments in bioinformatics, acquiring the complete set of features for every training instance is prohibitive, and it is impossible to wait for a complete set of features [9]. Besides, for the product to be processed in an industrial production line, it always requires multiple steps by different devices which dynamically generate different streaming features over time [10]. Online streaming feature selection that deals with feature streams in an online manner has attracted extensive attention recently [11].

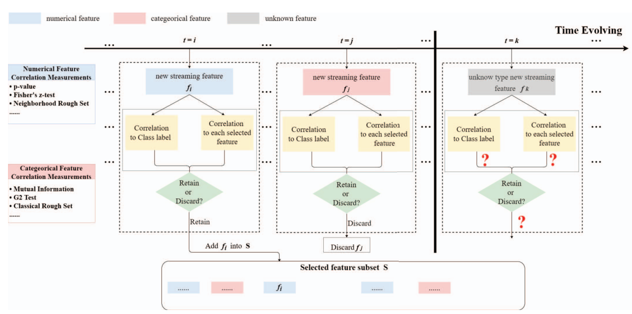


Fig. 1: Illustration of the problem of unknown type streaming feature selection. Streaming features are being generated and arriving one by one as time goes on (from t_1 to t_m). Usually, streaming feature selection methods need to measure the correlation between a new arriving feature f_i and the class label C, and the correlation between f_i and each feature f' in the selected feature subset S. However, if we cannot know the feature type of the next arriving feature, how can we measure the correlations?

Feature selection methods can be broadly categorized as the filter, wrapper, and embedded according to different selection strategies [12]. Unlike traditional feature selection methods, there are two main challenges for streaming feature selection: (1) the entire feature space is unknown or even infinite, (2) and we must decide whether to retain or discard the new arrival feature on the fly [13]. Due to storage space limitations, once a new arriving feature is discarded, we cannot use it again. Therefore, most existing online streaming feature selection methods apply a filter model to select the optimal streaming features [14]. In other words, these methods always need to design some measurements to calculate the association between features.

Generally speaking, the feature type of the target dataset can be categorized into categorical, numerical, or mixed. Existing streaming feature selection methods either design for single feature type or provide two versions of algorithms for both categorical and numerical features, respectively [11]. For

2375-9259/22/$31.00 ©2022 IEEE
DOI 10.1109/ICDMW58026.2022.00089

instance, based on penalized likelihood ratio, mutual information, and classical rough set theory, α-investing [15], GFSSF [16], and OS-NRRSARA-SA [17] are designed for categorical features respectively. In terms of neighborhood rough set theory, K-OFSD [18] and OFS-A3M [19] are proposed for numerical features only. Besides, based on statistical tests, information theory, and Fisher's Z-test, OSFS [8], SAOLA [20], SFS-FI [13], OSSFS-DD [21] provide two versions of algorithms for both categorical and numerical features respectively. For mixed feature space, fuzzy rough set-based methods [22], [23] or hybrid metrics based methods [24], [25] were proposed. All these methods mentioned above implicitly assume that we can know the attribute type of each feature before learning. However, it is unreasonable and unrealistic to know all the attribute types for the infinite streaming features in practical applications. As shown in Fig. 1, suppose at each timestamp t, the new arriving streaming feature is f_t. Filter model streaming feature selection methods usually use specific measurements to calculate the correlation between features. However, if we cannot know the feature type of the next arriving feature, how can we measure the correlations and decide whether to retain or discard this streaming feature? Motivated by this, this paper firstly studies a practical issue of online feature selection for the unknown type streaming features.

Specifically, we firstly pay attention to the issue of unknown type streaming feature selection and give a formal definition of it. Based on information theory, we model the streaming feature selection issue as a minimax problem and propose two metrics to determine whether the new arriving feature should be selected. Then we propose a new online feature selection method for unknown type streaming features, named UT-SFS. The main contributions of this paper are as follows:

- We first present the exciting and practical issue of unknown type streaming feature selection and model it as a minimax problem.
- In terms of MIC which can measure the correlation for unknown type features, we derive a new metric MIC_{Gain} that can be used to determine whether a new streaming feature should be selected. To speed up the efficiency of online feature selection, we present the metric MIC_{Cor} that can directly discard new arriving features with low correlation.
- We propose a new unknown type streaming feature selection method UT-SFS based on these two new metrics. UT-SFS is nonparametric and does not need to know the feature type of each streaming feature in advance, which is in line with practical application needs.
- Extensive experiments conducted on nineteen real-world datasets and compared with four state-of-the-art traditional mixed feature selection algorithms and five online streaming feature selection approaches indicate the effectiveness of UT-SFS.

The rest of this article is organized as follows. Section II describes related work. Section III presents the formal definition of the problem, the relevant theoretical knowledge of MIC, and a new method for unknown type streaming feature selection. Section IV gives the experimental analysis and Section V gives a brief conclusion.

II. RELATED WORK

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

A. Traditional Feature Selection Methods

According to the feature type of a dataset, feature selection methods can be divided into categorical, numerical, and mixed. Most traditional filter model feature selection algorithms are designed for a single feature type, i.e., categorical or numerical.

In practical applications, features may be gathered in mixed types. Therefore, some traditional mixed feature selection algorithms are proposed to deal with mixed feature space. Specifically, Zhang et al. [24] constructed a new information entropy measurement method based on fuzzy rough set theory for the mixed feature selection problem and proposed a new filter-wrapper model feature selection algorithm according to this measurement criterion. Yuan et al. [22] proposed the FRUAR algorithm for the feature selection problem of unsupervised mixed data. Yuan et al. [23] solved the feature interaction problem in the feature selection of unsupervised imbalanced mixed data and proposed a measure of uncertainty based on fuzzy complementary entropy, named EUIAR. For mixed feature type datasets, mixed feature selection methods use different metrics to decrease the information loss in the feature space. However, these methods require complete knowledge of the feature space before learning.

B. Online Streaming Feature Selection Methods

For some real-world applications, features may exist in a streaming model, and we cannot know the whole feature space before learning [6], [7], [9]. Therefore, many online feature selection methods have been proposed to solve the issue of streaming feature selection [11].

Specifically, Zhou et al. [15] proposed the Alpha-investing algorithm, which does not require a global model. However, Alpha-investing requires prior knowledge of the feature space structure to control the process of candidate feature selection heuristically. Wu et al. [8] proposed an online streaming feature selection framework, which includes two algorithms: OSFS and Fast-OSFS. Yu et al. [20] proposed the SAOLA method for high-dimensional data by using a pairwise comparison method based on mutual information theory. Rahmaninia et al. [26] used a streaming method to evaluate the correlation and redundancy of features based on mutual information theory and proposed two online feature selection algorithms, named OSFSMI and OSFOMI-k. Zhou et al. [13] proposed a streaming feature selection algorithm SFS-FI considering the interaction between features, and the number of selected features increased due to the consideration of the interaction ability between features.

Most existing streaming feature selection methods are designed for a single feature type or provide two versions of algorithms for both categorical and numerical features, respectively. However, besides the number of streaming features in practical applications, their feature type may also be unknown in advance. Therefore, this paper focuses on online streaming feature selection with unknown feature types.

III. THE PROPOSED METHOD

This section describes the formal definition of the problem and the specific implementation of the proposed method. We summarize some symbols used in this paper in Table I.

TABLE I: Summary on Mathematical Notations

Notations	Definition
D	Target dataset
F	Feature space
C	Class label
$\lvert\cdot\rvert$	lSl: the size of set S
x_i	i^{th} sample
f_j	j^{th} feature
U	Sample space: $\{x_1, x_2, ..., x_n\}$
S_t	The selected feature subset after time stamp t
$I(\cdot;\cdot)$	I(f;C): denote the mutual information between f and C
$MI(\cdot,\cdot,\cdot)$	$MI(D,k,l)$: denote the mutual information divided according to the integers (k,l) on the two-dimensional variable dataset D.

A. Problem Definition

Suppose F is the conditional feature space of the target dataset D, the class label is C, and the sample space is $U = \{x_1, x_2, ..., x_n\}$, where x_i is the i^{th} sample. For streaming feature selection, we cannot known the exact number of $|F|$ in advance (e.g. $|F| \rightarrow \infty$). At timestamp t, the new arriving streaming feature is f_t ($f_t \in F$), and we do not know the attribute type of f_t. Meanwhile, we must decide whether to retain or discard the new arrival feature on the fly, and the selected feature subset after timestamp t is S_t. Streaming feature selection aims to maximize the information of S_t at each timestamp while making the size of $|S_t|$ as small as possible.

Mutual information can measure the amount of information shared between S_t and C by measuring their dependency level. Therefore, in terms of information theory, online streaming feature selection can be formalized as:

$$min_{|S_t|} max\{I(S_t;C)\} \quad s.t. \quad |S_t| > 0 \tag{1}$$

Similar to traditional feature selection methods, two main issues for streaming feature selection can be distinguished: feature measurement and search strategy [27]. This first one is to define an appropriate measure function to calculate the correlation for each new arriving feature. The second issue is to develop a search strategy that can decide whether retain or discard each streaming feature. There are many measure functions, such as Pearson Correlation Coefficient (PCC) [28], Spearman's Rank Correlation Coefficient (SPCC) [29] and Mutual Information (MI) [30], etc. However, most existing feature measure functions must know the feature type before calculation. Therefore, first of all, we need a measure function to calculate the correlation between unknown type streaming features.

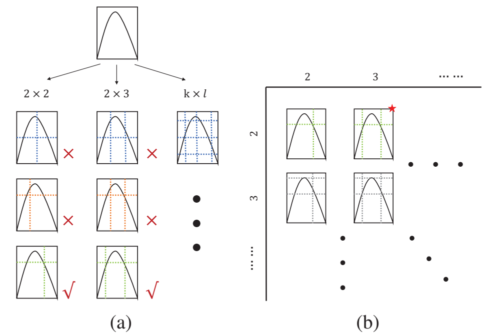


Fig. 2: Taking a parabola as an example, a schematic diagram of calculating MIC. (a) shows that for each pair (k,l), the MIC algorithm finds the k-by-l grid with the highest mutual information. (b) shows the maximum mutual information matrix $M(D)$ composed of the highest mutual information value obtained by each pair (k,l).

B. Measure Function for Unknown Type Features

MIC has been proved to be an effective measure of the dependence of two variables and can capture a wide range of both functional and unfunctional associations [31]. As shown in Fig.2, the x-axis and y-axis axes are divided dynamically in the calculation of the MIC. Therefore, MIC can calculate mutual information for both numerical and categorical data, making it adaptable to various applications. Specifically, given a two-dimensional variable dataset $D = \{(x_1,y_1),(x_2,y_2),...,(x_n,y_n)\}$. The integers (k,l) can be any pair. The calculation of the MIC(D) is as follows:

$$MIC(D) = max\{M(D)_{k,l}\} \tag{2}$$

$$M(D)_{k,l} = \frac{maxMI(D,k,l)}{logmin(k,l)} \tag{3}$$

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 $B(n) = n^{0.6}$.

MIC can measure the correlation between two variables of any type. A higher MIC value indicates a strong correlation between variables, and conversely, a lower MIC value implies a weak correlation between variables.

C. Seach Strategy for Streaming Features

Unlike traditional feature selection methods that actively search for optimal features, streaming feature selection can only passively receive streaming features and decide whether to retain or discard these features. At each timestamp, the ultimate goal of unknown type streaming feature selection is to maximize $MIC(S_t;C)$.

Metric MIC_{Gain}: Let $S = [f_1, f_2, ..., f_N]$ be an N dimensional feature vector and C is the class label. MIC measures the amount of information shared between S and C by measuring their degree of correlation. Denote the joint distribution densities of S and C and their marginal distributions by $P(S,C)$, $P(S)$, and $P(C)$, respectively. The MIC between features and class label can be defined as follows:

$$\begin{aligned} MIC(S;C) &= MIC(f_1, f_2, ..., f_N; C) \\ &= \int P(S,C) log \frac{P(S,C)}{P(S)P(C)} d_S d_C \end{aligned} \tag{4}$$

Although mutual information measurement [32] has good theoretical performance, accurate estimation of mutual information is impossible. Because to compute (4), the estimation of $P(S,C)$ is unavoidable, which is an NP-hard problem.

Suppose at timestamp t, the selected feature subset is S_t. It is impossible to calculate the information between a feature set S_t and a class label C directly [31]. Therefore, a more commonly used approach is to approximate it. To propose a new approximation, we formulate the unknown type streaming feature selection as:

$$max\{S_t^T Q_t S_t\} \tag{5}$$

where Q_t is a symmetric information matrix constructed from the mutual information terms in as:

$$Q_t = \begin{bmatrix} MIC(f_1;C) & ... & -\frac{\beta}{2}MIC(f_1;f_N) \\ -\frac{\beta}{2}MIC(f_1;f_2) & ... & -\frac{\beta}{2}MIC(f_2;f_N) \\ ... & ... & ... \\ -\frac{\beta}{2}MIC(f_1;f_N) & ... & MIC(f_N;C) \end{bmatrix} \tag{6}$$

where $S_t = [s_1, ..., s_N]$ is the selected feature vector, $s_i \in \{0, 1\}$, and β is a trade-off parameter.

At timestamp $t+1$, suppose the new arriving feature is f_{t+1}, and we add f_{t+1} into the candidate feature subset. That is, the selected feature subset is $S_{t+1} = [S_t, 1]$. If

$$S_{t+1}^T Q_{t+1} S_{t+1} > S_t^T Q_t S_t \tag{7}$$

then, f_{t+1} can be retained. Otherwise, we should remove f_{t+1} from S_{t+1}. Therefore, the condition for judging whether f_{t+1} should be selected is

$$S_{t+1}^T Q_{t+1} S_{t+1} - S_t^T Q_t S_t > 0. \tag{8}$$

In our proposed metric, the variable β is set to reciprocal of the number of selected features. Therefore, we define the metric MIC_{Gain} at timestamp t as follows:

$$MIC_{Gain}(f_t, S_{t-1}) = MIC(f_t;C) - \frac{1}{|S_{t-1}|} \sum_{f_i \in S_{t-1}} MIC(f_i;f_t) \tag{9}$$

The value of MIC_{Gain} determines the importance of newly arrived feature f_t to the currently selected subset S_{t-1} at timestamp t. If MIC_{Gain} is greater than 0, the newly arrived feature is positive for the complete information of the selected subset; otherwise, the value of MIC_{Gain} is less than 0.

Metric MIC_{Cor}: For streaming feature selection, the speed of the algorithm is critical. Because MIC needs to divide the variables into multiple grids, the time complexity of MIC is a bit high. Besides, in practical applications, there are always many irrelevant or low correlation features. Therefore, to speed up the online streaming feature selection, we propose a new metric MIC_{Cor} to discard these irrelevant and low correlation features directly.

$$MIC_{Cor}(S,C) = \frac{1}{|S|} \sum_{f_i \in S} MIC(f_i;C) \tag{10}$$

MIC_{Cor} is the mean correlation of each features in the currently selected feature subset. In other words, MIC_{Cor} aims to filter out low correlation features and maximize the correlation of the selected subset

$$max\{MIC_{Cor}(S_t,C)\}. \tag{11}$$

For a new arriving feature f_t, if $MIC(f_t;C)$ is samller than $MIC_{Cor}(S_{t-1},C)$, then it can be discarded directly.

Therefore, to maximize the correlation of the selected feature subset, we can discard the low correlation streaming features safely and directly in terms of MIC_{Cor}.

D. The Proposed Algorithm

To sum up, in terms of (9) and (10), we propose a new online streaming feature selection algorithm for unknown type streaming features as Algorithm 1.

More specifically, if a new feature f_t arrives at timestamp t, Steps 5-8 calculates the correlation values between f_t and C, then compares $MIC(f_t;C)$ to $Mean_S$, and selects the features with high correlation for the further evaluation processes. Steps 9-12 decide whether the newly arrived feature f_t is important for the candidate feature subset. If $MIC_{Gain}(f_t,S) > 0$, which mean the new feature f_t can increase the information of selected feature subset, we add f_t into subset S. With this new online streaming feature selection algorithm, we can select features with high correlation and high significance while ignoring the feature type of each streaming feature. Besides, it is worth mentioning that our algorithm does not need to set any parameters in advance.

E. Time Complexity

Here is an estimation of the time complexity of the algorithm UT-SFS. Let m and n be the numbers of features and samples for the target dataset, respectively. Because the MIC calculation uses a dynamic programming algorithm and

Algorithm 1 Unknown Type Streaming Feature Selection

Input:
 F: the condition feature set;
 C: the class attributes;
Output:
 S: the selected feature set;
1: **Initialization:**$S = \{\}$;
2: $MIC_{Cor}(S, C)$:the mean correlation of features in S, initialized to 0;
3: **Repeat**
4: Get a new arriving feature f_t at time stamp t;
5: IF $MIC(f_t; C) \leq MIC_{Cor}(S, C)$
6: Discard feature f_t;
7: Go to Step 13;
8: End IF
9: IF $MIC_{Gain}(f_t, S) > 0$
10: $S = S \cup \{f_t\}$;
11: End IF
12: **Until** no more features are available;
13: **Output** selected features contained in S.

the time complexity is difficult to determine. Therefore, we assume that the time complexity of MIC is constant $O(\Omega)$. At time stamp t, suppose that the number of selected features is $|S_t|$. The time complexity of steps 5-8 is $O(\Omega)$ and steps 9-12 is $O(m * |S| * \Omega)$. In sum, the worst time complexity of UT-SFS is $O(m^2\Omega)$ when we select all the streaming features. However, there are always many low correlation features for real-world datasets, and it is impossible for all features to increase the information of the selected feature subset. Thus, the time complexity of UT-SFS will be much smaller than $O(m^2\Omega)$.

IV. Experiments

A. Experimental Setup

1) Datasets: This section applies the proposed online streaming feature selection method (UT-SFS) and competing algorithms on nineteen real-world datasets. The details of these datasets are shown in Table II. Since the extremely long running time of the traditional mixed feature selection methods on high-dimensional datasets, the first five small datasets (German, Heart, Australian, Flags, Dermatology) are used to compare UT-SFS with four traditional mixed feature selection methods.

2) Evaluation Metrics: We use three base classifiers, KNN (k = 3), SVM (with the linear kernel), and CART in MATLAB, to evaluate selected subsets of features in our experiments. We perform a 5-fold cross-validation on each dataset. Feature selection is to train on 4/5 of the data samples and test on the remaining 1/5 of the samples. All competing algorithms use the same training and test sets. For each dataset, the order of stream features is random. We ran each dataset ten times and recorded the average prediction accuracy, running time, and the mean number of features selected on each classifier.

To verify whether the average prediction accuracy of UT-SFS and its competitors on different classifiers is significantly different, we performed the Friedman test at 95% significance level under the null hypothesis [33]. If the null hypothesis is rejected, there is a significant difference in the performance of UT-SFS and its competitors. When the null hypothesis of the Friedman test was rejected, we proceeded to the Nemenyi test as a post-hoc test [33].

3) Computational Device: All experimental results are conducted on a PC with AMD 5800X, 3.8 GHz CPU, and 16 GB memory.

TABLE II: Real-world Datasets

Data Set	instances	Features	Classes	Feature Type
German	1000	20	2	mixed
Heart	303	13	2	mixed
Australian	690	14	2	mixed
FLags	358	29	7	mixed
Dermatology	358	34	6	real
Arrarrhythmia	452	279	16	mixed
LYMPHOMA	62	4026	3	Real
SRBCT	63	2308	4	Real
DLBCL	77	6285	2	Real
CAR	174	9182	11	Real
OVARIAN	253	15154	2	Real
LEU	72	7129	2	Real
PROSTATE	102	6033	2	Real
ARCENE	200	10000	2	Real
LUNG2	203	3312	5	Real
LUNG	181	12533	2	Real
SYLVA	216	14394	2	mixed
GISETTE	7000	5000	2	Integer
DEXTER	600	20000	2	Integer

B. UT-SFS vs. Traditional Mixed Feature Selection Methods

In this section, we compare UT-SFS with four state-of-the-art traditional mixed feature selection methods including ε-approximate reduct [24], IFSM [25], EUIAR [23], and FRUAR [22]. All algorithms are implemented in MATLAB. Since the extremely long running time of these four algorithms on high-dimensional datasets, we only conduct the experiments on the first five small datasets as shown in Table II. The parameters involved in the comparison algorithms use the default values mentioned in the papers.

Tables III-VII summarize the predictive accuracy on different classifiers, the running time, and the mean number of selected features of these competing algorithms. The p-values of Friedman test on KNN, SVM, CART, running time and the mean number of selected features are 0.221e-05, 0.366e-05, 0.0038, 0.113e-09 and 0.0271 respectively. Thus, there is a significant difference between UT-SFS and the other four competing algorithms on predictive accuracy, running time, and the mean number of selected features. According to the Nemenyi test, the value of CD is 2.7294.

TABLE III: Predictive Accuracy Using KNN as the Classifier

Data Set	IFSM	ε-approximate	EUIAR	FRUAR	UT-SFS
German	0.6436	0.6981	0.613	0.5083	**0.7009**
Heart	**0.7519**	0.747	0.5478	0.5341	0.7241
Australian	0.7625	**0.8308**	0.4449	0.6194	0.8287
FLags	0.4098	0.3726	0.3742	0.3516	**0.5649**
Dermatology	0.8411	**0.9632**	0.3617	0.3475	0.9466
AVG.	0.6818	0.7222	0.4683	0.4722	**0.753**
AVG. RANKS	2.4	2	4	4.8	**1.8**

From Tables III-VII, we can observe that:

TABLE IV: Predictive Accuracy Using SVM as the Classifier

Data Set	IFSM	ε-approximate	EUIAR	FRUAR	UT-SFS
German	0.7	**0.7344**	0.6996	0.3897	0.7035
Heart	0.7837	**0.8107**	0.7056	0.4822	0.7563
Australian	0.7897	**0.8551**	0.4449	0.8191	**0.8551**
FLags	**0.4005**	0.3711	0.3366	0.2892	0.302
Dermatology	0.8651	**0.9595**	0.4539	0.2978	0.9407
AVG.	0.7078	**0.7462**	0.5281	0.4556	0.7115
AVG. RANKS	2.6	**1.3**	4	4.6	2.5

TABLE V: Predictive Accuracy Using CART as the Classifier

Data Set	IFSM	ε-approximate	EUIAR	FRUAR	UT-SFS
German	0.6277	0.6854	0.6922	0.5794	**0.7046**
Heart	0.747	**0.7848**	0.6974	0.6004	0.6974
Australian	0.761	**0.8475**	0.4464	0.7129	0.832
FLags	0.5007	0.4484	0.339	0.4346	**0.5428**
Dermatology	0.8612	**0.9316**	0.4419	0.8084	0.9111
AVG.	0.6995	**0.7395**	0.5234	0.6271	0.7376
AVG. RANKS	2.8	**1.8**	4.1	4.4	1.9

TABLE VI: Running time(seconds)

Data Set	IFSM	ε-approximate	EUIAR	FRUAR	UT-SFS
German	**0.1102**	2.7083	7.2998	342.2607	0.2638
Heart	**0.0041**	0.0574	0.1413	1.9774	0.1096
Australian	**0.0179**	0.6145	1.4561	105.0164	0.7406
FLags	0.0127	0.1685	1.8246	3.6457	**0.0122**
Dermatology	**0.0422**	0.5662	3.4394	21.4266	0.0747
AVG.	**0.03742**	0.823	2.8322	94.8654	0.2402
AVG. RANKS	**1.2**	2.6	4	5	2.2

TABLE VII: The mean number of selected features

Data Set	IFSM	ε-approximate	EUIAR	FRUAR	UT-SFS
German	9.04	11.58	3	16.2	2
Heart	4.46	6	3	11.66	5.22
Australian	6.62	6	3	12.76	7
FLags	7.88	9.28	3	6.86	1
Dermatology	8.2	17.96	3	13.66	20.52
AVG.	7.24	10.164	3	12.228	7.148
AVG. RANKS	2.8	3.8	1.4	4.2	2.8

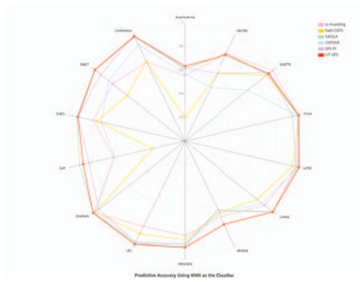
- UT-SFS $vs.$ IFSM: UT-SFS gets higher average predictive accuracy and lower average ranks than IFSM in cases of KNN, SVM, and CART. IFSM is faster than UT-SFS in running time and selects almost the same average number of features. IFSM is a neighborhood rough set-based incremental feature selection method to handle the dynamics of an object set that involves the change of a single object and multiple objects. Since the time complexity of the rough set model is square to the number of instances, IFSM is not capable of handling large datasets. Besides, IFSM needs to know the corresponding feature types before learning and can only handle static datasets.
- UT-SFS $vs.$ ε-approximate: There is no significant difference between UT-SFS and ε-approximate on predictive accuracy. The predictive accuracy of ε-approximate is slightly better than that of UT-SFS in cases of SVM and CART but worse in the case of KNN. ε-approximate is a supervised mixed feature selection algorithm based on fuzzy rough sets. ε-approximate can define corresponding fuzzy relationships for different features, which requires knowing the feature types before learning. Meanwhile, the time complexity of the ε-approximate is very high and unsuitable for processing high-dimensional datasets.
- UT-SFS $vs.$ EUIAR: UT-SFS performs better than EUIAR on predictive accuracy in cases of these three classifiers. Meanwhile, UT-SFS is faster than EUIAR in running time. EUIAR is an unsupervised mixed feature selection algorithm based on fuzzy rough sets and selects the fewest features that may lead to the loss of some critical information. Besides, EUIAR requires two thresholds to be given before feature selection to control the radius and the number of selected features. On the contrary, it is challenging to specify parameter values for streaming feature selection before learning.
- UT-SFS $vs.$ FRUAR: FRUAR performs the worst on predictive accuracy among all these competing algorithms. Meanwhile, there is a significant difference between UT-SFS and FRUAR in the case of KNN. FRUAR uses fuzzy rough sets to define the importance of individual features. The time complexity and space complexity of fuzzy rough sets based algorithms are very high. Therefore, the running time of FRUAR is much higher than other comparison algorithms.

In sum, UT-SFS is competing or better on predictive accuracy than these traditional mixed feature selection algorithms while does not need to know the type of each feature. Besides, UT-SFS is designed for high-dimensional datasets, while these traditional mixed feature selection algorithms cannot handle it due to exceptionally long running time.

C. UT-SFS vs. Online Streaming Feature Selection Methods

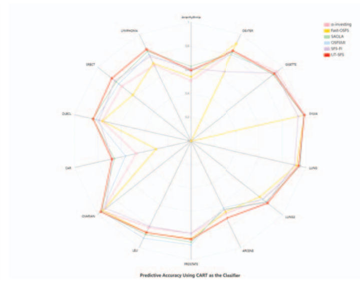
In this section, we compare UT-SFS with five state-of-the-art online streaming feature selection algorithms including α-investing [15], Fast-OSFS [8], SAOLA [20], OSFSMI [26], and SFS-FI [13]. We conduct the experiments on fourteen high-dimensional datasets as shown in Table II. Since most of these datasets are numerical features, we randomly selected 50% of the features and discretized these features into ten equal parts. Thus, all experimental datasets are mixed feature types for our new method. Meanwhile, because these five competing algorithms cannot handle mixed features, we use their categorical version algorithms in experimental, and the datasets are equidistantly discretized into two intervals. All algorithms are implemented in MATLAB. For α-investing, the parameters are set to the values used in [15]. The significance level α was set to 0.01 for Fast-OSFS and SAOLA, and the parameter value of SFS-FI was set to 0.05.

Fig. 3 summarizes the predictive accuracy on three different classifiers of these competing algorithms. Tables VIII-IX summarize the running time and the mean number of selected features. The p-values of Friedman test on KNN, SVM, CART, running time, and the mean number of selected features are 0.5584e-05, 0.3863e-10, 0.0015, 0.662e-14, and 0.704e-07 respectively. Thus, there is a significant difference between these competing algorithms on predictive accuracy, running time and number of selected features. According to the Nemenyi test, the value of CD is 2.015. Fig. 4 shows the statistical test of these competing algorithms in cases of KNN, SVM, and CART.



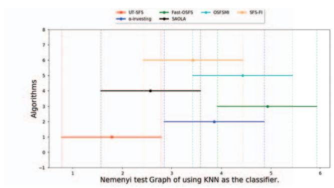
(a) KNN

(b) SVM

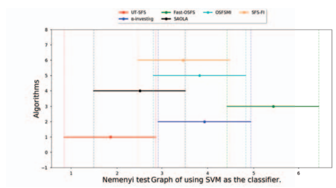


(c) CART

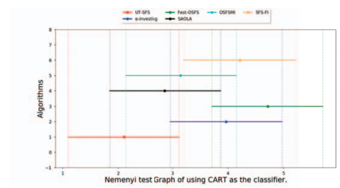
Fig. 3: Predictive accuracy of these competing algorithms



(a) KNN



(b) SVM



(c) CART

Fig. 4: The statistical test graph of these competing algorithms

TABLE VIII: Running time(seconds)

Data Set	α-investing	Fast-OSFS	SAOLA	OSFSMI	SFS-FI	UT-SFS
Arrarrhythmia	**0.0077**	0.2624	0.0246	0.2613	0.1044	8.3131
LYMPHOMA	**0.0651**	2.3755	0.897	0.461	3.294	40.1677
SRBCT	**0.0286**	1.1187	0.2155	0.2059	4.3432	0.8469
DLBCL	**0.1387**	2.9935	0.2323	1.0585	1.2055	13.4415
CAR	**0.5461**	7.1761	6.3329	1.2339	241.2094	46.9425
OVARIXAN	1.3392	14.6532	**0.9837**	12.7006	13.8304	44.3455
LEU	**0.195**	3.4568	0.2511	0.7164	21.8869	16.0849
PROSTATE	**0.1318**	2.9869	0.172	0.6658	0.6187	6.9797
ARCENE	0.4637	5.4661	**0.2295**	124.8731	1.2078	54.5846
LUNG2	**0.1436**	3.2084	2.2243	0.4246	7.8362	132.6171
LUNG	**1.0266**	8.4292	1.7554	1.7922	3.4119	43.809
SYLVA	0.2746	132.0478	**0.0735**	3.4589	0.1433	114.0287
GISETTE	58.9935	386.4437	**1.1407**	778.7647	16.4794	849.6236
DEXTER	2.4291	8.7344	**0.3374**	2092.413	1.3195	20.5311
AVG.	4.6988	41.3823	**1.0621**	215.645	21.2065	99.4511
AVG. RANKS	**1.5714**	4.6429	1.9286	3.5	3.7857	5.5714

TABLE IX: The mean number of selected features

Data Set	α-investing	Fast-OSFS	SAOLA	OSFSMI	SFS-FI	UT-SFS
Arrarrhythmia	5.56	3	21.32	100.14	80.22	21.78
LYMPHOMA	6.04	2	166.52	17.08	240.64	319.46
SRBCT	6.62	2	56.4	9.98	664.98	20.8
DLBCL	11.24	2.06	60.7	25.12	51.52	142.54
CAR	24.16	2	308.06	9.4	6042.7	109.8
OVARIXAN	32.92	2.96	32.82	73.48	207.68	45.52
LEU	16	2	43.82	7.18	77.72	164.6
PROSTATE	10	2.14	22.74	8.3	21.42	47.96
ARCENE	10.08	3.02	27.08	2232.44	22.64	35.88
LUNG2	20.12	3	322.42	12.1	432.22	170.1
LUNG	34.38	3.2	283.38	9.22	52.78	96.46
SYLVA	37.48	14.44	9.64	95.72	2.64	16.9
GISETTE	297.98	10.14	20.58	1882.28	48.94	70.76
DEXTER	12.74	2.1	32.2	15024.46	22.24	87.7
AVG.	37.5229	3.8614	100.5486	1393.35	594.8814	96.4471
AVG. RANKS	2.9231	1.1538	4.0769	3.6923	4.3077	4.8462

From Figs. 3-4 and Tables VIII-IX, we can indicate that:

- UT-SFS $vs.$ α-investing: According to the statistical test results, UT-SFS performs significantly better than α-investing on predictive accuracy in cases of KNN and SVM. Besides, UT-SFS gets much high predictive accuracy than α-investing on most of these datasets by using CART as the classifier. The running time of α-investing is the shortest among these competing algorithms. However, α-investing does not handle redundancy between features and select few features on sparse datasets.
- UT-SFS $vs.$ Fast-OSFS: There is a significant difference in predictive accuracy between UT-SFS and Fast-OSFS in cases of KNN, SVM, and CART. Fast-OSFS performs the worst on predictive accuracy among all these competing algorithms. On running time, Fast-OSFS is a little faster than UT-SFS. Fast-OSFS select the fewest features that may lead to the loss of important information and result in lower prediction accuracy.
- UT-SFS $vs.$ SAOLA: According to statistical tests, UT-SFS and SAOLA have no significant difference in predictive accuracy. UT-SFS gets higher predictive accuracy on average and lowers average ranks than SAOLA. SAOLA is faster than UT-SFS due to its pairwise comparison method. Meanwhile, they select about the same number of features on these datasets. Like UT-SFS, SAOLA also uses mutual information to select features on the fly but can only deal with single-type streaming features.
- UT-SFS $vs.$ OSFSMI: UT-SFS performs significantly better than OSFSMI on KNN. In cases of SVM and CART, UT-SFS gets higher predictive accuracy on average and lowers average ranks than OSFSMI. On running time, OSFSMI is speedy on some datasets but spends the most time on other datasets. OSFSMI selects the most features on average among these competing algorithms. Thus, the performance of OSFSMI varies widely on different datasets, which indicates its poor adaptability.

- UT-SFS $vs.$ SFS-FI: UT-SFS performs significantly better than SFS-FI on CART. In cases of KNN and SVM, UT-SFS gets higher predictive accuracy on average than SFS-FI. SFS-FI is faster than UT-SFS in running time. Since SFS-FI considers feature interaction, it selects more features on some datasets. Like UT-SFS, SFS-FI also uses mutual information to select features but cannot handle mixed features and unknown type features.

In sum, UT-SFS achieves the highest predictive accuracy and lowest ranks among these competing algorithms on these datasets. Besides, since UT-SFS is nonparametric and does not need to know the feature type of each streaming feature in advance, it is better in line with practical application needs.

V. CONCLUSION

In this paper, we propose a novel online streaming feature selection method to address the issue of unknown type streaming features, which is more in line with practical applications. We model the issue of unknown type streaming feature selection as a minimax problem. In terms of MIC, which can measure the correlation between any feature, we derive two new metrics that aim to select informative and compact features. Extensive experiments demonstrate the effectiveness of our new proposed method compared to four traditional mixed feature selection algorithms and five online streaming feature selection methods. However, the time complexity of UT-SFS is a bit high due to the calculation of MIC, and we will focus on how to reduce the running time in future work.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under grants (61906056, 61806002).

REFERENCES

[1] H. Liu and 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, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, 2018.

[3] S. Solorio-Fernández, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "A review of unsupervised feature selection methods," *Artificial Intelligence Review*, vol. 53, no. 2, pp. 907–948, 2020.

[4] E. Hancer, B. Xue, and M. Zhang, "A survey on feature selection approaches for clustering," *Artificial Intelligence Review*, vol. 53, no. 6, pp. 4519–4545, 2020.

[5] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing*, vol. 300, pp. 70–79, 2018.

[6] M. Wang, H. Li, D. Tao, K. Lu, and X. Wu, "Multimodal graph-based reranking for web image search," *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4649–4661, 2012.

[7] W. Ding, T. F. Stepinski, Y. Mu, L. Bandeira, R. Ricardo, Y. Wu, Z. Lu, T. Cao, and X. Wu, "Subkilometer crater discovery with boosting and transfer learning," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 4, pp. 1–22, 2011.

[8] X. Wu, K. Yu, W. Ding, H. Wang, and X. Zhu, "Online feature selection with streaming features," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 5, pp. 1178–1192, 2013.

[9] J. Wang, P. Zhao, S. C. Hoi, and R. Jing, "Online feature selection and its applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 698–710, 2013.

[10] M. H. u. Rehman, E. Ahmed, I. Yaqoob, I. A. T. Hashem, M. Imran, and S. Ahmad, "Big data analytics in industrial iot using a concentric computing model," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 37–43, 2018.

[11] X. Hu, P. Zhou, P. Li, J. Wang, and X. Wu, "A survey on online feature selection with streaming features," *Frontiers of Computer Science*, vol. 12, no. 3, pp. 479–493, 2018.

[12] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[13] P. Zhou, P. Li, S. Zhao, and X. Wu, "Feature interaction for streaming feature selection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, pp. 4691–4702, 2021.

[14] N. AlNuaimi, M. M. Masud, M. A. Serhani, and N. Zaki, "Streaming feature selection algorithms for big data: A survey," *Applied Computing and Informatics*, 2020.

[15] J. Zhou, D. P. Foster, R. A. Stine, and L. H. Ungar, "Streamwise feature selection," *Journal of Machine Learning Research*, vol. 3, no. 2, pp. 1532–4435, 2006.

[16] H. G. Li, X. D. Wu, Z. Li, and W. Ding, "Group feature selection with streaming features," in *IEEE 13th International Conference on Data Mining*, 2013, pp. 1109–1114.

[17] S. Eskandari and M. Javidi, "Online streaming feature selection using rough sets," *International Journal of Approximate Reasoning*, vol. 69, no. C, pp. 35–57, 2016.

[18] P. Zhou, X. Hu, P. Li, and X. Wu, "Online feature selection for high-dimensional class-imbalanced data," *Knowledge-Based Systems*, vol. 136, pp. 187–199, 2017.

[19] P. Zhou, X. Hu, P. Li, and X. Wu, "Online streaming feature selection using adapted neighborhood rough set," *Information Sciences*, vol. 481, pp. 258–279, 2019.

[20] K. Yu, X. Wu, W. Ding, and J. Pei, "Scalable and accurate online feature selection for big data," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 11, no. 2, pp. 1–39, 2016.

[21] P. Zhou, S. Zhao, Y. Yan, and X. Wu, "Online scalable streaming feature selection via dynamic decision," *ACM Trans. Knowl. Discov. Data*, vol. 16, no. 5, pp. 1–20, 2022.

[22] Z. Yuan, H. Chen, T. Li, Z. Yu, B. Sang, and C. Luo, "Unsupervised attribute reduction for mixed data based on fuzzy rough sets," *Information Sciences*, vol. 572, pp. 67–87, 2021.

[23] Z. Yuan, H. Chen, and T. Li, "Exploring interactive attribute reduction via fuzzy complementary entropy for unlabeled mixed data," *Pattern Recognition*, vol. 127, p. 108651, 2022.

[24] X. Zhang, C. Mei, D. Chen, and J. Li, "Feature selection in mixed data: A method using a novel fuzzy rough set-based information entropy," *Pattern Recognition*, vol. 56, pp. 1–15, 2016.

[25] W. Shu, W. Qian, and Y. Xie, "Incremental feature selection for dynamic hybrid data using neighborhood rough set," *Knowledge-Based Systems*, vol. 194, p. 105516, 2020.

[26] M. Rahmaninia and P. Moradi, "Osfsmi: online stream feature selection method based on mutual information," *Applied Soft Computing*, vol. 68, pp. 733–746, 2018.

[27] T. Naghibi, S. Hoffmann, and B. Pfister, "A semidefinite programming based search strategy for feature selection with mutual information measure," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 8, pp. 1529–1541, 2014.

[28] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise reduction in speech processing*. Springer, 2009, pp. 1–4.

[29] A. Kumar and S. Abirami, "Aspect-based opinion ranking framework for product reviews using a spearman's rank correlation coefficient method," *Information Sciences*, vol. 460, pp. 23–41, 2018.

[30] J. R. Vergara and P. A. Estévez, "A review of feature selection methods based on mutual information," *Neural computing and applications*, vol. 24, no. 1, pp. 175–186, 2014.

[31] D. N. Reshef, Y. A. Reshef, H. K. Finucane, S. R. Grossman, G. McVean, P. J. Turnbaugh, E. S. Lander, M. Mitzenmacher, and P. C. Sabeti, "Detecting novel associations in large data sets," *science*, vol. 334, no. 6062, pp. 1518–1524, 2011.

[32] Y. W. Lee, "Statistical theory of communication," *American Journal of Physics*, vol. 29, no. 4, pp. 276–278, 1961.

[33] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.