IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 2 February 2023, accepted 8 March 2023, date of publication 16 March 2023, date of current version 23 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3257857

RESEARCH ARTICLE

Online Active Learning Framework for Data Stream Classification With Density-Peaks Recognition

KUANGYAN ZHANG, SANMIN LIU, AND YANFEI CHEN
School of Computer and Information, Anhui Polytechnic University, Wuhu 241000, China

Corresponding author: Sanmin Liu (sanmin.liu@ahpu.edu.cn)

This work was supported by the University Natural Science Research Project of Anhui Province under Grant KJ2021A0516 and Grant 2022AH050972.

ABSTRACT This paper considers a practical scenario — data stream classification, and uses online active learning (OAL) to assist the classifier. Most active learning are proposed based on label uncertainty, but the methods based on representativeness are faced with difficulties in stream scenarios. The sampling criteria based on representativeness are usually fixed, which makes it difficult for them to work effectively in dynamic sample spaces, especially for concept drift streams. This paper devises a novel OAL framework based on sample representativeness. The framework uses local nearest-neighbor relation to measure the representativeness of unlabeled samples and divides the maximum influence space of the representative sample. We also develop an independent mechanism to identify and store short-term cluster fragments to ensure the integrity of information. The framework is deployed to the online environment and adapts to any incremental learner. Simulation experiments on multiple datasets and classifiers show that our method can help primary classifier deal with data stream environment effectly. Compared with other online active learning, the method can achieve more stable accuracy and better anti-noise ability under fewer budget.

INDEX TERMS Data stream classification, online active learning, density-peak clustering, cluster fragmentation.

NOMENCLATURE

t	Time stamp.
x_t	Instance at time t.
y_t	Ture label of instance x_t.
ds	Data stream.
B	Budget for label cost.
f	Classifier or learner for active learning.
C	Cluster based on density reachability.
$BatchSize$	Size of cache latest instances.
N_{x_i}	Set of nearest neighbors for x_i.
A_{x_i}	Number of neighbor association for x_i.
h	Altitude threshold.
x'	Tagged peak sample.
N'_{x_i}	Sample set of influence range for x_i.
R_{x_i}	Neighbor influence range for x_i.
S_E	Extended instance set.
R_C	Representative instance set in C.
$CluFra_t$	Cluster fragmentation set at time t.

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Wei Tsai.

I. INTRODUCTION

With the increasing scale and complexity of network, data stream mining has already been the focus of machine learning [1], [2]. It is widely used in practical domains such as signal processing [3], social media [4], abnormal stream detection [5], and other scenes. Unlike static machine learning, data stream owns the characteristics of high-speed, continuous, changeable, infinite, and so on [6], [7]. Because of these characteristics, the distribution of data constantly changing in a potential and uncertain way. This phenomenon

is usually referred to as concept drift [8]. For streams with variable distribution, constructing an adaptive classifier is a crucial topic in data streaming mining, and it is of great significance in machine learning algorithms and experimental processes [9].

There are many unlabeled samples in real-time data and sufficient manually labeled data are challenging to meet, this causes the insufficient labeled instances for current pattern recognition systems [10], [11]. For example, in the industrial IoT, the classifier detects and diagnoses faults through sensor data, the difficulty lies in the annotation of each event by experienced workers, and this process is usually time-consuming and laborious [12]. In intrusion detection system, data must be processed efficiently to maintain a secure network environment. Due to the large number and scale of network connections, manual labeling of all connection data is expensive and impractical. Therefore, the system must perform detection and update accurately and efficiently under conditions of label scarcity. In similar scenarios, selection and labeling samples under a limited budget have well utility in building well-performance classifiers, so the application of active learning and unsupervised learning have become important topics.

Human learning paradigm inspires active learning: Firstly, purposefully explore areas that have not yet been learned; Secondly, optimize the learner by querying the sample label to achieve self-optimization and long-term learning [13]. In order to select samples with high value, information (also known as discriminativeness) query and representativeness query are the current mainstream means for selecting unlabeled samples [10], [11], [16]. The first is to select the information-rish samples to reducing the potential classifiers as soon as possible [14], [15]. The second is to select representative samples and reduce the redundancy between training samples [17], [18]. Most strategies seek representative samples by constructing the input pattern for overall data. However, the performance of the methods is reduced in the data stream environment because the preset data pattern changes at any time.

As an effective unsupervised learning method, density-based clustering (DBC) can identify the distributed characteristics of sample space [15], [19]. However, traditional DBC algorithms rely heavily on the constraint: each candidate cluster has only one density-peak. In a variable data stream environment, a cluster may have 0 or more density-peaks, and peaks are surrounded by samples with lower density. The distance between peaks with high density is relatively significant. Therefore, peaks can not be distinguished under the fixed distance of the neighbor [20]. Driven by the above considerations, our cognition is to mine highly local information and find potential density-peaks in the cluster, representing other samples of local space to the greatest extent.

Unlike static data clustering, clustering fragments are difficult to handle in incremental clustering and data stream mining [16]. They usually appear, grow and disappear in the evolution process of the stream. The traditional strategies with representativeness query only focus on the regions with dense distribution and consider the instances in the sparse area as outliers. Therefore, the design leads to information loss and clustering fragmentation in the sparse region. If the cluster fragment is a component of an important concept that needs to be learned in the future, or more samples are added to the cluster fragment over time, and more complete concepts will be formed. Our cognition is to save short-term fragmentary clustering and learn it when the concept becomes mature.

According to the above expound, the objective of our strategy is to improve the traditional sampling method with representativeness query. We developed an altitude measurement method and a searching representativeness instance method to effectively achieve instances that can fully represent the whole community. We integrate the methods into the data stream classification, called Online Active Learning with Density-peaks Recognition (OAL-DpR). In the training process, the classifier adapts to the concept drift by passively and incrementally updating. In addition, the OAL-DpR also adds a method to identify and save potential information (clustering fragmentation), which is helpful to mining dynamic data steam. Generally, the contributions are as follows:

1) Online active learning with representativeness: To solve the limitations of traditional sampling strategy with representativeness in data stream environment, we first apply the density-peaks recognition method to online active learning and remeasure the information accuracy of instances in local regions;
2) Information retention mechanism: We recognize the incomplete clusters (fragmentation) in the dynamic sample space through the altitude of these instances, and set up a separate memory space to store the pieces of information and query the growth of these in the next iteration;
3) A novel scheme for data stream learning: In this article, we develop a batch-incremental scheme and apply the density-peak recognition task to online machine learning to solve the lack of representativeness query in the existing active learning.

The organization of paper is shown as follows: Section II expounds the related works about OAL and DBC. The OAL method with density-peaks recognition is presented in Section III. The experiment scheme and result are discussed in Section IV. Finally, drawing the conclusion in Section V.

II. RELATED WORK

As far as we know, OAL-DpR is the first algorithm that combines density-peak clustering with online active learning to solve the label-scarcity stream classification. Given the gap between the related fields, we introduce the research progress of the two areas separately in this section.

A. ONLINE ACITVE LEARNING

Facing the limitation of label-scarcity in data stream classification, the current mainstream research directions are semi-supervised learning, transfer learning and active learning. Semi-supervised learning does not rely on external interaction and automatically uses unlabeled data to improve learning performance and avoid waste of data resources. Transfer learning tries to solve the target domain contains only a small number of labeled samples by using the existing knowledge in different fields. Active learning was first proposed by Angluin of Yalu University to solve the label scarcity in machine learning [13]. The uncertainty is the most critial research object in the early period, such as the distance between the decision boundary and samples [14], [21], information entropy, and expected error [22], [23]. Later, scholars joined the samples with representativeness to the training set as a supplement to the uncertainty [17], [18], [21], [24], [25].

Most algorithms query the input sample patterns through clustering or conditional filtering and select the instances that maximize the classifier's generalization. These algorithms not only need to deal with the scarcity of labels but also solve the concept drift and outliers that invariably affect the stability of learners due to changes in the data collection environment and equipment losses. The different query engines are designed to purposefully measure the information value and representativeness of samples, then label the samples with valuable [6], [10], [11], [16].

Bougueria et al. [6] believed that the uncertain query threshold of the query engine should also be adjusted when the sample space of the stream changes. The threshold was reduced when the concept drift occurred and remained high for the rest of the time. The integrated learning framework based on active sampling is proposed in [10]. The sampling engine gradually reduces the uncertainty threshold when the distribution drifts. It prioritizes querying the most uncertain instances and dynamically allocates the query cost. Mohamad et al. [11] proposed bi-criteria query strategies: label-uncertainty and density-based standard. The second criteria show the actual distribution by weighting the labeled samples to alleviate the problem of sampling deviation. Zhang et al. [14] think that when the decision boundary of the time sequence changes, the boundary points in the dense area reflect the actual data distribution to a great extent. Zhang puts forward an OAL method with detecting boundary and outlier based on [11], which combines the sample space information, and theoretically alleviates the problem of sampling deviation.

In recent research, pool-based active learning methods have been gradually introduced into online active learning [7], [15], [25], [26]. These methods assist sampling by assuming a priori distribution of the sample space or preclustering, which reduces the impact of outliers the sampling bias caused by uncertainty strategies to some extent. An online active sampling method based on kNN is proposed in [15]. This method defines the local uncertainty and chooses instances in each batch. To achieve sampling in unbalanced class data, a double-layer mixed labeling strategy is proposed in [7]: uncertainty sampling is implemented based on information margin and adjusted threshold, then the instances of minority category are learned from the latest data blocks using the imbalance strategy. An evolutionary approach (ESBMAL) is proposed in [20]. The active learning task was modeled as a multi-objective problem that is solved by a genetic algorithm. Reference [26] selects valuable instances iteratively in each data block with uncertainty value, label correlation, and label sparsity.

In a word, most of the studies are concentrated in the following ways: (1) quantifying the uncertainty of classifier on sample information; (2) reducing the impact of sampling deviation through prior knowledge of distribution; (3) selecting the representative samples by measuring the accuracy of local spatial information.

B. DENSITY-PEAK CLUSTERING

DBSCAN is one of the most influential algorithms of DBC [27]. It derives clusters connected by the relationship with maximum density reachability. Therefore, DBSCAN can cluster dense data sets of arbitrary shape only using two appropriate parameters: ϵ and MinPts. Being insensitive to noise and outliers is the most practical feature of DBSCAN. However, DBSCAN has limited effect in data with varying density-distribution (VDD) or multiple domain-density maximum (MDDM) [16], [25]. Samples will be misallocated to adjacent clusters when constant Delta distance are used, which leads to the loss of sample information and cluster fragmentation. Recently, the study based on density-peak has made a lot of improvements on the basis of DBC.

The authoritative density-peak algorithm is Clustering by Fast Search and Finding of Density Peaks (CFSFDP), which was proposed by Rodriguez et al. [8]. His core idea is that the cluster center is surrounded by samples with lower density, and the distance between the peak is relatively prominent. CFSFDP detect arbitrary shape clustering in low complexity and non-iterative way. The concept of Fuzzy Density Peaks Clustering was developed by Bian et al. [30]. It expresses the density of points as the coupling of the fuzzy distance between data and adjacent data. Cheng et al. [28] proposed a Minimum Spanning Tree based on Local Density-peak (LDP-MST). LDP-MST first organizes all data into a directed acyclic graph, then converts the root nodes in the directed acyclic graph into a minimum spanning tree according to the shared neighborhood distance between the root nodes. After removing the unnecessary edges, each minimum spanning tree is considered a cluster. A faster LDP-MST method is proposed in [29], which is not conducive to running on high-dimensional and large-scale data.

For data distribution with varying densities or changes constantly, the DBC algorithm faces the limitations of sparse information loss and cluster fragmentation. Building an adaptive strategy based on neighbor selection is a common way to solve this problem. Chen [16] proposes Domain Adaptive

Density Clustering (DADC), which uses adaptive density measurement to extract cluster peaks and merge fragment clustering. DADC includes three steps: domain adaptive density detection, automatic cluster center recognition and automatic cluster integration. Du et al. [31] calculated local density based on kNN and principal component analysis, and proposed a high-dimensional data-intensive region recognition method. Kamali et al. [32] proposed the density clustering method of neighborhood distance entropy consistency (NEDC) and took NEDC as an essential criterion for merging potential subclusters.

III. ONLINE ACTIVE LEARNING FRAMEWORK FOR CLASSIFICATION WITH DENSITY-PEAKS RECOGNITION

To solve the lack of strategies for selecting representative samples in OAL, the paper proposes a method for data stream classification called OAL-DpR. The whole framework is shown in Fig.1. It realizes iterative update of the classifier through four steps.

Step 1: Two-layer clustering learning. The spatial neighbor information of the batch samples is mined through preclustering with DBSCAN and the Measurement Method of Sample Altitude (MMSA). The peak samples in the cluster are selected to provide priori-knowledge for processing each cluster;

Step 2: Multi-peak cluster identification. The goal is to classify clusters with different peaks. Each category performs various operations: the cluster fragments are stored in the following iterative learning, and the clusters containing the cluster peaks are further processed in Step 3.

Step 3: Search for representative samples and knowledge extraction. An efficient representative sample search method (RSSM) is proposed in this process. This method reduces the redundancy between the extracted samples and the data pool using the distribution characteristics of spatial objects;

Step 4: Incremental training of the classifier. Oracle labels the representative samples and submits them for incremental training. The classifier trained with representative samples has a more vital generalization significance.

In the main body of this section, we first introduce some preliminary knowledge. Then in Section III-B, we give the details of Step 1, including the definition of neighbor association and density-peak in this paper. Next, in Section III-C, we introduce the multi-density-peak clustering identification in Step 2 and the RSSM in Step 3. Then we discuss how OAL-DpR works in the online environment. Finally, we describe the key parameters and complexity analysis.

A. PRELIMINARIES

The paper defines the stream ds as a set of continuous data with infinite length:

$$ds = \{(x_1, y_1), (x_2, y_2), \ldots, (x_i, y_i), (x_{i+1}, y_{i+1}), \ldots\} \quad (1)$$

where $x_i = \{a_1^i, a_2^i, \ldots, a_l^i\}$is a eigenvector of instance at time i, l is the dimension of the feature vector. $y_i = \{1, 2, \ldots, c\}$ is the label collection, c is the number of sample labels.

In the data stream classification task, most of the sample labels y_i cannot be obtained directly. Therefore, the goal is to select samples from the pool (batch mode) for labeling under limited annotation resources. Traditional active learning usually set budget B to control the number of samples. B is generally charged as a percentage, the ratio of the labeled to candidate samples. OAL-DpR selects valuable information through spatial distribution knowledge without setting label preselection in advance. Therefore, OAL-DpR need not specify the $\hat{B}$ for annotation in advance to affect the whole sampling result.

B. MEASUREMENT METHOD OF SAMPLE ALTITUDE

Most DPC methods are based on three perfect hypotheses: 1. density peak is arounded by the samples with lower density; 2. there may be multiple peaks in a cluster; 3. there is a significant distance between different density peaks. For the samples x_i, its local density ρ_i can be defined:

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \quad (2)$$

where d_c is the cut-off threshold, if $d_{ij} - d_c < 0$, $\chi(d_{ij} - d_c) = 1$, else $\chi(d_{ij} - d_c) = 0$. Therefore ρ_i means the number of samples that are close to x_i. Another feature of x_i is the Delta distance δ_i. By calculating the shortest distance between x_i and others samples with more significant density, δ_i is defined as:

$$\delta_i = min_{j:\rho_j > \rho_i} d_{ij} \quad (3)$$

For the maximum-density sample, $\delta_i = max_j d_{ij}$. The samples with high ρ and δ are peak, and the remaining data will be classified into the nearest cluster. Therefore, clustering effect of DPC is primarily determined by the calculation method of density peak. Like the density-based clustering algorithm, d_c is the variable that needs to be initially set. It is uncertainty in the VDD data, and the best d_c is usually fuzzy. To avoid the influence due to the uncertainty of preset parameters, the concept of sample altitude is proposed in this section.

For the convenience of description, we let N_{x_i} as the set of the neighbors for x_i in cluster C. Neighbor recognition is based on Euclidean distance.

a: DEFINITION 3.1 (NEIGHBOR ASSOCIATION NUMBER)

Given x_i as a sample in C, its neighbor association number A_{x_i} is the number that x_i belongs to the N_{x_j} of any other sample x_j in C. The formula is described as follows:

$$A_{x_i} = \sum_{x_j \in C} \Pi(x_i \in N_{x_j}) \quad (4)$$

In (4), Π is the indicator function that outputs 1 if the inequation is true, or else outputs 0. A_{x_i} reflects the number of samples with x_i as a neighbor in the cluster.

The expression of neighbor association number is similar to the altitude in geography: the samples in the cluster have different heights due to the degree of spatial density. To make

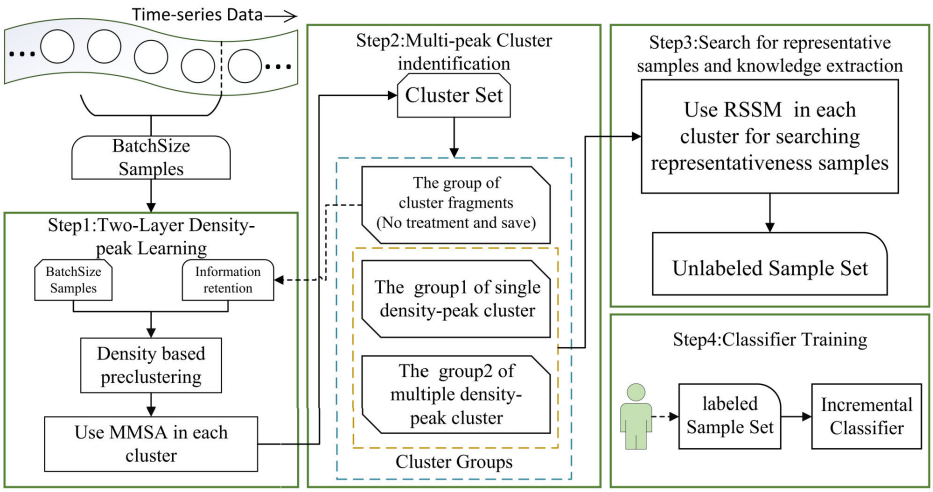

FIGURE 1. Workflow of online active learning based on density-peak recognition.

the method more effective in showing the difference in altitude in the cluster, we hope that the scale of the cluster will not be too small. Note that the nearest-neighbor relationship in this paper is asymmetric. The number of N_{x_i} for x_i depends on the priori distribution of spatial objects. Therefore, there is a case that one sample becomes the neighbor of multiple samples, and there is a case that one sample is not the neighbor of other samples.

The steps of the measurement method based on the neighbor association are shown in Fig.2. Each sample in C has been numbered in order. First, we traverse each sample x_i to get its nearest-neighbor set N_{x_i}. When the scale is appropriate, it's sufficient to set the size of N_{x_i} to 2. Next, the information of each sample is retained in the form of two tuples. The elements of the two tuples are the serial number and sample altitude, defined above. Finally, we use different colors to represent samples. The darker the color, the greater the altitude value of the sample. The sample altitude measurement method based on asymmetric nearest-neighbor is described as Algorithm 1.

Algorithm 1 Measurement Method of Sample Altitude

1: **require:** density-peak cluster C;
2: **ensure:** two-tuples (i, A_{x_i}) of each sample in C;
3: **for** each $x_i \in C$ **do**
4: researching for the two nearest-neighbors N_{x_i};
5: statistical the times of x_i was nearest neighbor;
6: output (i, A_{x_i});
7: **end for**

C. REPRESENTATIVE SAMPLE SEARCH METHOD BASED ON LOOP SEARCH PEAK

In the active learning stage, we aim to find an optimal classifier that can generalize the unknown data well. Furthermore, to avoid the classifier being retrained after new samples are added to the label set, it is necessary to consider empirical risk in sampling, which minimizes the upper bound of active learning risk constraints [17]. The upper bound can be approximated by the sum of the difference between of the discriminative and the adequately designed regularization term.

For a given distribution $p(x, y)$, the goal of active learning is to train a classifier $f \in \mathcal{F}$ which is based on a lower expected risk $\mathcal{R}(f)$, $\mathcal{F}$ is the function space. $\mathcal{R}(f)$ is given as follow:

$$\mathcal{R}(f) = \mathbb{E}_{(x,y)\sim p(x,y)} L(f(x), y) \tag{5}$$

with $L(f(x), y)$ is loss on train data. *Wang* and *Ye* [23] provide active learning risks related to expected risks based on the empirical risks of the query sample set: given a kernel function $K(x_i, y_i)$, it uses a nonlinear relationship $\phi(x)$ for mapping samples to kernel Hilbert space Q.

A labeled set S with size n and Q with size q which are mixed with labeled and unlabeled samples, $q \ll n$, then the probability of following is over $1 - \tau$:

$$\mathcal{R}(f) \leq \hat{R}_Q(f) + MMQ_\phi(S, Q) + C(\mathcal{F}, q, \tau) \tag{6}$$

with $MMQ_\phi(S, Q)$ is the empirical estimation term of S, $C(\mathcal{F}, q, \tau)$ is term for complex terms of functions. In the description of the risk boundary of active learning, the second item limits difference in distribution between the labeled and all samples, it's as similar as possible. In other words, the higher the representativeness of S, the small the value of this term and the more stringent the risk of active learning.

a: DEFINITION 3.2 (PEAK SAMPLE)

The peak sample x_p is the sample with the highest altitude in C and the A_{x_p} exceeds h, where h is the altitude threshold. Other samples whose altitude exceeds the threshold are potential peak samples.

A stable cluster can be described as a community, and a community can have multiple representatives. We hold that most samples in the cluster can be represented by the peak samples in the cluster. We set the altitude threshold h of the

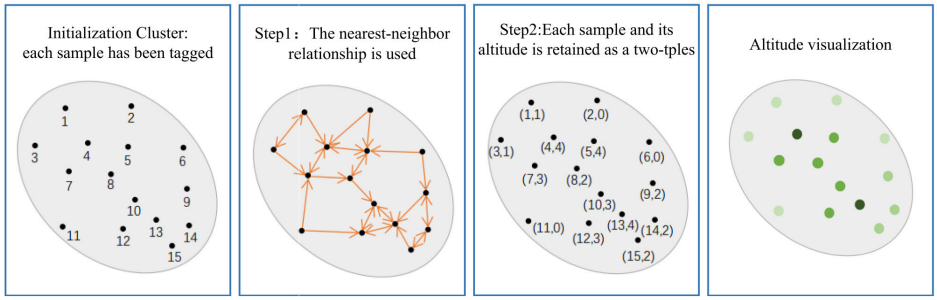

FIGURE 2. An illustration of the measurement method of sample altitude.

lowest peak sample to obtain a better representation of peak samples. If the highest altitude in cluster C is less than h, the highest sample can't be the peak sample.

To express the influence range of peak samples, we propose the influence range of samples and define clustering fragments. We will further propose a representative sample selection method based on loop peak search.

In Section III-B, we obtain the altitude of each sample. If there is a peak sample in the C, mark it as x_p. We set the altitude threshold h of the lowest peak sample to obtain better representativeness of peak samples. If the highest altitude A_C in C is less than h, the sample with the highest altitude can't be marked as a peak sample.

b: DEFINITION 3.3 (SAMPLE SET OF INFLUENCE RANGE)

Given a sample x_i in C, N'_{x_i} is the influence range set of x_i, which can be described: for any sample $x_k \in C$, if $x_j \in C$, and $\{x_i, x_k\}$ is the subset of N_{x_j}, $x_k \in N'_{x_i}$. N'_{x_i} is the nonempty subset of the set C.

c: DEFINITION 3.4 (NEAREST-NEIGHBOR INFLUENCE RANGE)

Given a sample x_i in C, the nearest-neighbor influence range R_{x_i} of x_i is the space which containing only N'_{x_p} and x_p.

The Fig.3 shows the influence range sample set N'_A and the nearest neighbor influence area R_A of point A. Due to the 2 nearest-neighbor sample set N_B of point B is $\{A, C\}$, so C is included in N'_A. Similarly, point B, D, E is also included in N'_A. N_F and N_G include point A, so F and G are included in N'_A. In summary, $N'_A = \{B, C, D, E, F, G\}$, and samples that only include N'_A and point A is the nearest-neighbor influence range R_A, which is marked with red dotted line.

Through the above analysis, we can find the spatial range region R_{x_i} for any sample x_i. While x_i is closer to the density center of the cluster, the spatial range R_{x_i} is larger, so we choose the peak sample x_p in C as a representative sample to cover the influence range sample set N'_{x_p} of x_p can't cover all samples in C. Therefore, we query whether there is a subpeak sample x'_p in the remaining sample set C', where $C' = \complement_C N'_{x_p}$. It should be a local or overall community space. However, in most scenarios, the noted that $x'_p \cap N'_{x_p} = \emptyset$, x'_p is the subpeak sample of C that meets the altitude threshold. Before finding the next subpeak sample x'_p each time, it is necessary

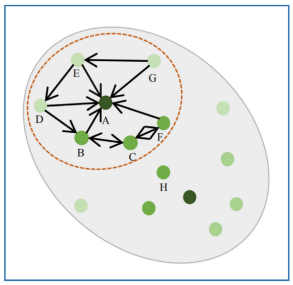

FIGURE 3. Example of sample set of influence range and nearest-neighbor influence range.

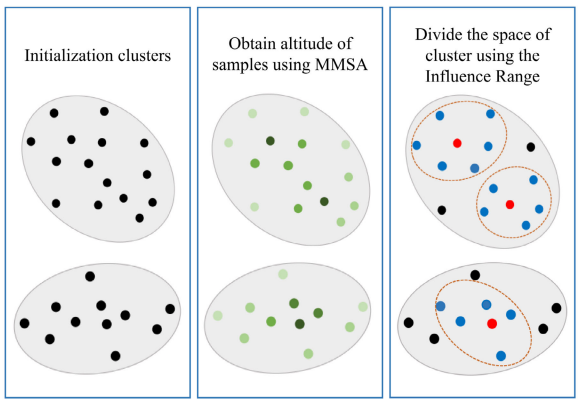

FIGURE 4. Identification process of single density-peak cluster and multi density-peaks cluster.

to remove the obtained peak sample, subpeak sample, and their influence range sample set from C. Different from the traditional strategy with representativeness sampling to select the center or the edge of the cluster, we divide the sample cluster into one or more ranges and scattered samples.

We label the density-peak cluster with no subpeak samples as a single-density-peak cluster and the cluster with subpeak as a multi-density-peak cluster. Fig.4 shows the identification process of single-density-peak and multi-density-peak clusters. We propose a representative sample search method (RSSM) by searching iteratively. Fig.5 illustrates the operation of the methods. The result of MMSA is used as the starting condition of RSSM, which makes it convenient for us to search for peak samples quickly. The whole process is to explore the remaining space which is not affected by the representative samples in the cluster.

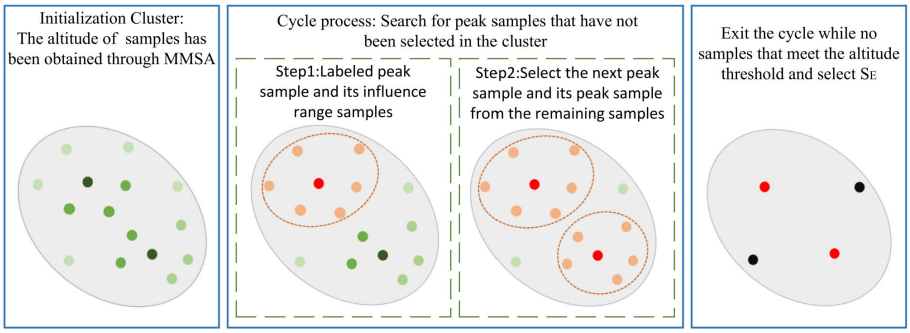

FIGURE 5. An Illustration of the representativeness sample search method.

First, search for x_p that meets the altitude threshold h while having the highest altitude in C. Then, search for x_p's influence range sample set N'_{x_p}, and mark x_p as a representative sample. In the remaining sample space C', query whether there is a subpeak x'_p in C'. Find x'_p's influence range sample set $N'_{x'_p}$. We mark x'_p as a representativeness sample. Then get C'' by removing $N'_{x'_p}$ from the sample set C' and continue to find out whether there are subpeak samples that meet the altitude threshold in the remaining sample set C''. If no samples meet the altitude threshold, exit the search process. For the samples set C_s not selected into the scope of influence, based on historical experience, select the two samples with the largest distance from C_s, and add them to the representativeness sample set. And these samples are called extended samples S_E. In the follow-up experiments, we compared the experimental results on whether or not to add S_E. Our historical experience can bring better experimental results and a more significant labeling budget to the machine learning process. Algorithm 2 gives the process of RSSM with searching iteratively.

Algorithm 2 Representative Sample Search Method

require: C:density-peak cluster;
ensure: R_C: representative sample set of C;
if (peak sample x_p in C) **do**
  search for x_p's influence range sample set N'_{x_p} in C;
  add N'_{x_p} to R_C;
  $C' = \complement_C N'_{x_p}$
  while($\exists$ subpeak samples x'_p in C') **do**
    search the sample set $N'_{x'_p}$ of x'_p in C';
    $C' = \complement_C N'_{x'_p}$
    add s'_p to R_C;
  end while
end if
select S_E to add to R_C in C';

D. ONLINE ACTIVE LEARNING FRAMEWORK

Sampling strategies with sample similarity usually select instances in the cluster's center or edge. However, the design will face bottlenecks when processing multi-peak clusters or variable density distribution. RSSM rebalances the distinction and representativeness by measuring the similarity and differences from the attributes of samples. In this section, we apply MMSA and RSSM to the data stream classification task and describe how to deal with clustering fragmentation.

a: DEFINITION 3.5 (CLUSTER FRAGMENTATION)

x_p is highest in cluster C, if $A_{x_p} < h$, C is called clustering fragmentation.

To maintain the short-term concept, we set up an independent space named information retention to save the fragmented information. The retention will be added to the data pool in the next iteration. The growth of incomplete information is queried by maintaining the spatial objects in two periods, and MMSA and RSSM are used to extract the samples with annotation value query. The information retention of the previous iteration will be freed after this round of iteration.

Algorithm 3 describes the process of OAL-DpR. The algorithm realizes a round of iterative cycles of classification tasks in four steps. The processing of new samples in the time-series data uses batch-based learning to store short-term historical samples. Add the latest sample x_t in the time series to the current batch. If the number of samples in the batch reaches threshold, start a round of iterative active learning sampling process.

E. COMPLEXITY ANALYSIS

The time and space complexity of OAL-DpR is analyzed in this section, respectively.

1) TIME COMPLEXITY

Generally speaking, OAL-DpR consists of 3 parts: classifier prediction label, querying for representative samples, and classifier update. Let's make the following notation: T_p is the time for predicting the latest samples' label, T_u is the time for updating classifier with labeled label instances, T_q is the query time per iteration. Let n be the size of stream, α is the actual dimension scale, therefore, the overall time complexity can be approximate to $O(n * \alpha(T_u + T_q) + n * T_p)$.

2) SPACE COMPLEXITY

Since OAL-DpR uses batch processing to store the latest samples, the cache space is freed after each iteration. At the same time, we have opened up a space for information

Algorithm 3 Online Active Learning With Density-Peak Recognition

require: ds: data stream; h: altitude threshold; $BatchSize$: size of batch;
ensure: acc: accuracy of classifier;
initialize classifier f;
for each $x_i \in ds$ **do**
  f predict the label of x_i;
  add x_t to the current batch S_t;
  if($S_t.size == BatchSize$) **do**
    $S_t = S_t \cup CluFra_{t-1}$
    precluster to obtain the initial clusters;
    Algorithm 1 is used in each cluster;
    add clusters without peak to the $CluFra_t$;
    Algorithm 2 is used to obtain the set R_{S_t}
    Oracle label all samples in R_{S_t};
    f.incremental training(R_{S_t});
    $CluFra_t$.clear;
  end if
end for

retention, assuming the size is $IRSize$. Let S_f be size of space to store classifiers. Therefore, the overall space complexity approximates to $O(IRSize + BatchSize + S_f)$.

IV. EVALUATION EXPERIMENT

Evaluation of OAL-DpR proposed in the section, and the evaluation criteria are as follows: OAL-DpR can reduce the label budget while the classifier performance is stable; the accuracy of the classifier trained by OAL-DpR is within an acceptable range compared with supervised learning and other active learning algorithms. The experiment is divided into three stages. Firstly, the performance of OAL-DpR is compared with standard active learning algorithms on multiple data sets and classifiers. Then, the sensitivity of fundamental parameters is analyzed: *BatchSize* and altitude threshold h. Finally, the robustness of OAL-DpR is discussed based on noisy data. OAL-DpR and comparison algorithms are realized in the Massive Online Analysis (MOA) software [33], it's an open platform for time series wining. All evaluation processes are performed in MOA and Eclipse, we recorded the accuracy and budget of each process.

A. EXPERIMENTAL SETUP

1) EXPERIMENTAL SCHEME

To reduce the deviation caused by classifier category in the process of evaluating the applicability, HoeffdingTree [9], PaireLearners [34], OzaBagAdwin [35], NaiveBayes, and ADACC [34] are used in this paper. ADACC, PaireLearners, and OzaBagAdwin are ensemble learning classifiers. The classifier is updated after each iteration and predicts the new samples in the next iteration.

2) DATASET DESCRIPTION

We set up synthetic datasets and four real datasets. The description for each dataset is shown in Tab.1.

TABLE 1. Description of each dataset.

Dataset	#Instances	#Classes	#Attributes	#Noise ration
$Hyperplane_1$	10000	2	10	5%
$Hyperplane_2$	10000	2	8	10%
FCT	581012	7	54	-
$Electricity$	45312	2	8	-
$Phoneme$	5404	2	5	-
$WFRN$	5456	4	24	-

The synthetic dataset is generated based on the moving hyperplane, and the concept of time change is simulated through the transformation of hyperplane direction and position in two-dimensional space. A sequence of instances are generated from the equation $\sum_{i=0}^{l} a_i x_i = a_0$, where $a_0 \sum_{i=1}^{l} a_i$ and l suggests the number of dimensions.When $\sum_{i=0}^{l} a_i x_i \geq a_0$,the class label of instance $(x_1, x_2, \ldots, x_l)$ is $+1$ and -1 otherwise. We have created two datasets with different number of attributes, and add 5% and 10% noise samples respectively.

Forest Coverage Type (FCT) contains the actual forest coverage type of $30m * 30m$ cells obtained from the resource information system of the United States Forest Service ($USFS$).

Electricity is the electricity price change of an electricity market in New South Wales over a period of time. All instances are distributed in two categories. Each instance has 8 attributes and category labels indicating the rise and fall of electricity prices.

Phoneme is a set of phoneme sample collected by a sound sensor, which is used to distinguish between nasal sound (Class 0) and oral sound (Class 1).

Wall-Following Robot Navigation ($WFRN$) is collected when SCITOS G5 robot passes through the room clockwise along the wall. Each sample contains 24 attribute values obtained by the sensor around its waist, as well as 4 categories that judge the robot's action as forward, slight turn to right, sharp turn or slight turn to left.

B. SIMULATION EXPERIMENT

To evaluate the effect of OAL-DpR, the standard online active learning methods are selected to compare with OAL-DpR in sampling ratio and classification accuracy of the classifier. The strategies involved in the comparative experiment are Fixed Uncertainty (FU), Variable Uncertainty (VU), Randomly Variable Uncertainty (RVU) [37], and Selective Sampling (SS) [37]. The active learning strategies of comparison use the training method of batch processing. These strategies select sample query labels with a fixed ratio in each iteration and use them for incremental learning, and the percentage is set to 15%. To more conveniently evaluate the experimental results of each comparison strategy, we add supervised learning (SL) to the comparison experiment. The altitude threshold h in the OAL-DpR is set to 4 and the *BatchSize* is set to 100. The number of initialization set is 1% of the total samples.

The comparison of accuracy on the five classifiers is shown in Tab.2. In each line, the two with the highest accuracy are

TABLE 2. Comparison of accuracy on experiment datasets(%).

Dataset	Learner	SL	OAL-DpR	FU	VU	RVU	SS
$Hyperplane_1$	NaiveBayes	**91.85**	80.19	50.23	**87.69**	87.35	87.96
	HoeffdingTree	**88.19**	80.54	50.23	**87.93**	86.98	87.21
	ADACC	**84.20**	71.90	50.23	**81.80**	80.57	80.62
	OzaBagAdwin	**89.78**	81.95	50.23	**88.06**	87.58	86.36
	PairedLearners	**79.88**	68.47	70.65	71.90	72.22	**74.06**
$Hyperplane_2$	NaiveBayes	**87.14**	76.08	49.64	**83.83**	83.45	83.42
	HoeffdingTree	83.51	76.08	49.64	**83.83**	83.45	**84.17**
	ADACC	**78.48**	67.28	49.64	75.27	75.85	**76.07**
	OzaBagAdwin	**84.46**	74.58	49.64	**83.74**	83.20	83.54
	PairedLearners	**74.66**	63.44	49.64	66.69	**73.62**	73.60
FCT	NaiveBayes	**60.52**	60.50	48.8	**60.52**	60.51	60.34
	HoeffdingTree	**80.31**	**73.65**	1.65	69.83	70.21	70.59
	ADACC	**90.71**	**82.71**	50.97	80.41	80.21	79.28
	OzaBagAdwin	**84.74**	73.56	1.65	**79.00**	77.20	77.41
	PairedLearners	**89.03**	75.33	48.80	**78.73**	78.69	76.61
$Electricity$	NaiveBayes	73.33	73.14	57.52	**74.02**	**73.48**	73.51
	HoeffdingTree	**79.17**	74.14	58.22	**77.42**	75.65	76.12
	ADACC	**89.62**	**83.54**	46.46	81.05	80.49	80.79
	OzaBagAdwin	**84.34**	**80.50**	57.55	79.35	76.51	78.04
	PairedLearners	**87.13**	80.50	48.20	**80.80**	80.50	80.14
$Phoneme$	NaiveBayes	75.28	**77.88**	70.85	**77.27**	76.01	75.50
	HoeffdingTree	**77.34**	**77.9**	70.85	75.08	76.10	75.81
	ADACC	**75.09**	**78.51**	70.85	74.22	74.13	73.42
	OzaBagAdwin	**78.51**	**77.84**	70.85	76.68	76.21	75.4
	PairedLearners	**74.92**	**77.88**	70.85	73.87	74.77	72.04
$WFRN$	NaiveBayes	54.95	55.30	58.11	**60.27**	**60.27**	60.01
	HoeffdingTree	54.95	52.87	16.00	60.02	**86.98**	**87.21**
	ADACC	**80.53**	**68.05**	57.25	66.95	65.72	61.64
	OzaBagAdwin	74.66	55.30	16.56	88.06	**87.58**	**86.36**
	PairedLearners	**80.69**	**73.30**	48.51	64.33	66.73	62.95
Rank.Avg		1.76	3.76	5.93	2.75	3.37	3.71
Rank.Var		1.49	2.87	0.13	1.49	0.65	1.28

shown in bold and the ones with the highest accuracy are underlined. Rank.Avg and Rank.Var are the average rank and rank variance of the algorithms, respectively. Tab.4 shows the sampling ratio of OAL-DpR on each data set. OAL-DpR sampling is only related to the dataset and the fundamental parameters in OAL-DpR. The ratios are not affected by the classifier category and are unchanged in each dataset. From the results, the classification accuracy obtained by SL is the best, but other strategies also receive acceptable classification results. OAL-DpR proposed by us is a more effective method of all strategies. It is also less than different strategies compared with *SL* accuracy error. It shows that our representation-based sampling method is helpful to active learning. The results in Tab.4 show that the overall sampling ratio of OAL-DpR is also lower than the sampling ratio of the comparison algorithm, and the average labeling ratios are lower than 15%.

According to the results in Tab.2, FU is the most unstable strategy among comparison strategies. It leads to the collapse of classifiers on large-scale data sets such as *FCT* and *Electricity*, which is also the disadvantage of a single strategy. The improved RVU and VU can better adapt to the dynamic space, which also shows that it's not enough to only rely on the classifier to select the labeled samples. From the results of comparison, the classification of VU and RVU are similar and better than FU. As an effective method in random sampling algorithms, SS uses decision boundary analysis and perceptron to determine the current instance selected for annotation. The results show that SS can achieve a good classification effect.

Due to representativeness sampling havng low requirements for initial training, it is more suitable for online environments than uncertain sampling when initialization is insufficient. However, when the sampling strategy of the latter can select high-information instances, a classifier based on these samples often performs the effect closer to the supervised classifier. The results also showed that the classifier trained with SL is not always the best, for example, in *Phoneme* and *WFRN*. The classifier trained with an active learning strategy is better than supervised learning. Generally speaking, the classifier based on active learning has better generalization ability in unknown fields with less training samples. Due to the correlation between samples, training more meaningless samples won't necessarily bring more benefits to the classification model. On the contrary, it may cause overfitting of the classifier. Therefore, we need to take note of data distribution when dealing with continuous and messy data distribution.

C. ABLATION EXPERIMENT

In Algorithm 2, the extended dataset S_E is selected from the remaining samples when the iteration cycle exits and adds to the representativeness of samples. This process is based on historical experience. To prove the effectiveness, the classification results of $OAL-DpR'$ without extended sample S_E are compared on the same classifier and dataset. The initial samples are set to 1% of the total samples, h is set to 4, and *BatchSize* is set to 100. The experimental results obtained by

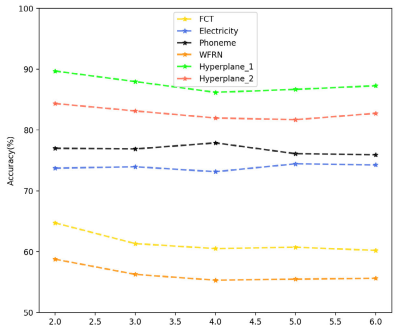

FIGURE 6. Effect of *h* change on classification accuracy.

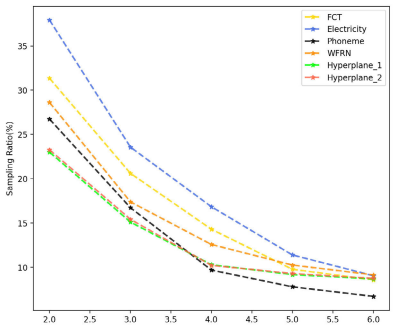

FIGURE 7. Effect of *h* change on sampling ratio.

training different classifiers with OAL-DpR and $OAL-DpR'$ are shown in Tab.3, and Tab.4 shows the sampling ratio of OAL-DpR and $OAL-DpR'$ on each dataset.

From the results, OAL-DpR is better than $OAL-DpR'$. The gap in classification accuracy is between 5% and 10% in most datasets. In *Phoneme*, the gap in accuracy is only 1% to 3%. According to Tab.4, the sampling proportion decreased by 20% to 40% in the experimental datasets after removing S_E, about 20% in the *FCT* with a large number of multi-density peak clusters, and 30% in the hyperplane synthetic dataset.

S_E is selected from the remaining sample set which is not included in the influence range in each cluster, and two samples are selected at most. Therefore, three samples will be selected as representative samples from a single-density-peak cluster, and $n+2$ samples will be selected from n density-peaks cluster. Obviously, OAL-DpR will discard fewer samples when the proportion of multi-peak clusters is high in datasets. The sampling proportion of $OAL-DpR'$ will drop to 1/3 of OAL-DpR when dealing with a single density-peak cluster. In general, the OAL-DpR can improve the classification accuracy compared to $OAL-DpR'$ but increase the budget of labeling samples.

D. PERFORMANCE EVALUATION

1) SENSITIVITY ANALYSIS

In this section, NaiveBayes is used as a classifier for sensitivity analysis with key parameters *h* and *BatchSize*.

h : As shown in algorithm 2, the lowest altitude threshold can be selected as the peak sample in any cluster. Set the *BatchSize* to 100, and increase *h* from 2 to 6. Fig.6 shows the accuracy of the classifier trained with OAL-DpR on the experimental data set, and Fig.7 shows the sampling ratio.

With the gradual increase of *h*, the variation range of accuracy in each dataset is about less than 5%, and the sampling ratio decreased on a slippery slope, dropping sharply in the field of 2 to 4. Although the sampling ratio is relatively high at the beginning, the accuracy of the classification model is not significantly superior. It shows that training more samples with low representativeness can't effectively improve the performance of the classifier, increasing the annotation budget. The higher altitude threshold can filter out a large number of samples with low representativeness, so as to obtain a classifier with excellent performance at a low cost.

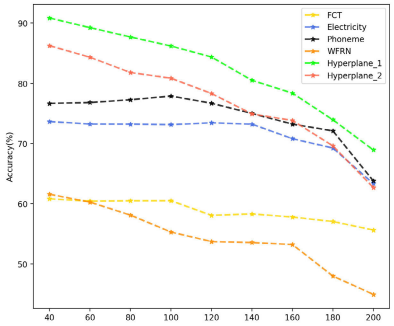

FIGURE 8. Effect of *BatchSize* change on classification accuracy.

BatchSize: It's the size of the cache space for the sampling engine to save the time-series data. We set the *h* to 4 and gradually increased the *BatchSize* from 40 to 200. Fig.8 shows the accuracy of the classifier trained with OAL-DpR on the experimental data set, and Fig.9 shows the sampling ratio.

Summarise from Fig.8 and 9 that the sampling ratio decreases continuously in varying degrees with *BatchSize* increases: the sampling proportion on *FCT* and *WFRN* is about 20%, while 10% on other datasets; In Fig.9, the sampling proportion also shows a significant downward trend. When the cache space is ample, the number of iterations will be reduced, which is the main reason for the continuous decline of sampling proportion. At the same time, with the increase of cache space, the update of the classifier lags behind the change of sample space, so the classifier's performance decreases in varying degrees on each dataset. However, the small cache space will increase the budget for active learning. It is not easy to find appropriate representative samples through the nearest-neighbor relationship.

2) ROBUSTNESS ANALYSIS

This section verifies the robustness of OAL-DpR on noisy datasets through experiments. The experimental dataset uses *SeaGeneration* in *MOA* to generate 15 datasets, and the noise ratio of each dataset ranges from 1.0% to 15.0%. The strategies compared are *VU*, *RVU*, and *SS*; NaiveBayes is used as the classifier; The *h* of OAL-DpR is set to 4, and the *BatchSize* is set to 100. We prepared a boxplot to explore the accuracy

TABLE 3. Accuracy of DpR and *DpR′* on each dataset(%).

Dataset	NaiveBayes		OzaBagAdwin		HoeffdingTree		PairedLearners		ADACC	
	DpR	DpR'	DpR	DpR'	DpR	DpR'	DpR	DpR'	DpR	DpR'
$Hyperplane_1$	80.19	77.40	81.95	76.89	80.54	77.09	68.47	65.09	71.90	67.16
$Hyperplane_2$	76.08	74.86	74.58	72.14	76.08	74.57	63.44	61.4	67.28	64.62
$Covtype$	60.50	56.47	73.56	67.43	73.65	68.29	75.33	68.04	82.71	74.33
$Electricity$	73.14	67.80	80.50	73.97	74.14	66.10	80.50	75.29	83.54	66.56
$Phoneme$	77.88	75.28	77.84	76.28	77.90	75.24	77.88	74.99	78.51	73.88
$WFRN$	55.30	49.86	55.30	52.11	52.87	47.86	73.30	65.23	68.05	59.41

TABLE 4. Sampling ratio of DpR and *DpR′*on each dataset(%).

Strategy	$Hyperplane_1$	$Hyperplane_2$	FCT	$Electricity$	$Phoneme$	$WFRN$
DpR	10.22	12.03	14.29	16.83	9.68	12.57
DpR'	6.46	8.87	11.62	11.79	6.77	9.18

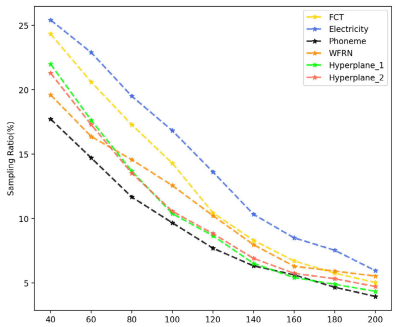

FIGURE 9. Effect of *BatchSize* change on sampling ratio.

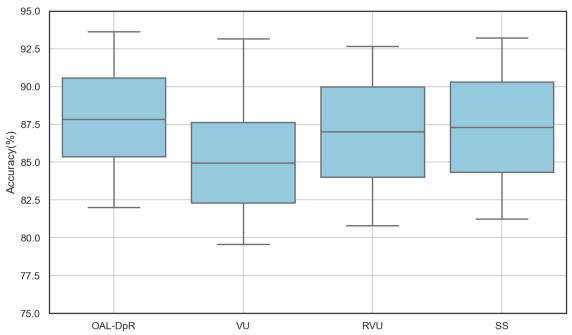

FIGURE 10. Boxplot for different methods in datasets with different ratio of noise.

changes of the four active learning strategies on 15 datasets. Fig.10 shows the experimental result.

Summarise from the boxplot that OAL-DpR has the most robust and stable anti-noise ability of all active learning strategies. On the other hand, the *VU* strategy is the worst in the comparison algorithm at each position. The active learning method based on the density-peak can effectively eliminate outliers by measuring the local density to reduce the impact of noise samples on the sampling engine. Therefore, the classification model based on OAL-DpR training has better generalization ability and better robustness to noise data. And the method based on uncertainty will be affected by noise samples when detecting the sample information. Due to the environment's instability and the acquisition equipment's aging, the noise can not be avoided. Therefore, the models with more robust to noise are usually more attractive.

V. CONCLUSION

An novel OAL method based on density-peak recognition is proposed in this work. As far as our knowledge, it's the first time that density-peak has been introduced into the online classification task. First, the information accuracy of the sample in clusters is determined by the asymmetric nearest-neighbor relationship, and representativeness samples are iteratively selected in each cluster by dividing the influence range of samples. In addition, OAL-DpR identifies and captures the fragmented information by dynamically saving short-term cluster fragmentation. Finally, in the framework of online batch mode, multiple classifiers and datasets are used for simulation experiments. By comparing the existing and commonly used strategies, OAL-DpR has the advantages of high stability, accuracy, and robustness. To a certain extent, it reduces the lack of a strategy for selecting representative samples in the current active learning task.

For the future work, we will further discuss the adaptive and incremental learning of machine learning under the conditions of low sample annotation ratio and complex environmental changes.

REFERENCES

[1] J. Gama et al., "Knowledge discovery from data streams," *Intell. Data Anal.*, vol. 12, no. 1, pp. 251–252, May 2008, doi: 10.3233/IDA-2008-12301.

[2] Y. M. Wen, L. Shuai, M. Yu-Qing, Y. Xin-He, and L. Chang-Jie, "Survey on semi-supervised classification of data streams with concept drifts," (in Chinese), *J. Softw.*, vol. 33, no. 4, pp. 1287–1314, 2022. [Online]. Available: http://www.jos.org.cn/jos/article/abstract/6476

[3] S. Mohamad, M. Sayed-Mouchaweh, and A. Bouchachia, "Online active learning for human activity recognition from sensory data streams," *Neurocomputing*, vol. 390, no. 1, pp. 1287–1314, May 2020, doi: 10.1016/j.neucom.2019.08.092.

[4] H. Y. Wang, X. G. Hu, and P. P. Li, "Semi-supervised short text stream classification based on vector representation and label propagation," *Pattern Recognit. Artif. Intell.*, vol. 31, no. 7, pp. 634–642, Jul. 2018, doi: 10.16451/j.cnki.issn1003-6059.201807006.

[5] B. Gou, Y. Xu, Y. Xia, G. Wilson, and S. Liu, "An intelligent time-adaptive data-driven method for sensor fault diagnosis in induction motor drive system," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9817–9827, Nov. 2018, doi: 10.1109/TIE.2018.2880719.

[6] X. Zheng, P. Li, Z. Chu, and X. Hu, "A survey on multi-label data stream classification," *IEEE Access*, vol. 8, pp. 1249–1275, 2020, doi: 10.1109/ACCESS.2019.2962059.

[7] H. Zhang, W. Liu, J. Shan, and Q. Liu, "Online active learning paired ensemble for concept drift and class imbalance," *IEEE Access*, vol. 6, pp. 73828–373815, 2018, doi: 10.1109/ACCESS.2018.2882872.

[8] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, Jun. 2014, doi: 10.1126/science.1242072.

[9] G. Hulten, L. Spencer, and P. Domingos, "Mining time-changing data streams," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, Aug. 2001, pp. 97–106.

[10] J. Shan, H. Zhang, W. Liu, and Q. Liu, "Online active learning ensemble framework for drifted data streams," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 486–498, Feb. 2019, doi: 10.1109/TNNLS.2018.2844332.

[11] S. Mohamad, A. Bouchachia, and M. Sayed-Mouchaweh, "A bi-criteria active learning algorithm for dynamic data streams," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 74–86, Jan. 2018, doi: 10.1109/TNNLS.2016.2614393.

[12] J. He, Q. Yang, and Z. Wang, "On-line fault diagnosis and fault-tolerant operation of modular multilevel converters—A comprehensive review," *CES Trans. Electr. Mach. Syst.*, vol. 4, no. 4, pp. 360–372, Dec. 2020, doi: 10.30941/CESTEMS.2020.00043.

[13] D. Angluin, "Queries and concept learning," *Mach. Learn.*, vol. 2, no. 4, pp. 319–342, Apr. 1988, doi: 10.1007/BF00116828.

[14] Y. Zhang, H. Yu, G. Wang, and Y. Xie, "An active learning method for adaptive classification of data streams," (in Chinese), *J. Nanjing Univ. Natural Sci.*, vol. 56, no. 1, pp. 67–73, Jan. 2020, doi: 10.13232/j.cnki.jnju.2020.01.008.

[15] H. Gweon and H. Yu, "A nearest neighbor-based active learning method and its application to time series classification," *Pattern Recognit. Lett.*, vol. 146, pp. 230–236, Mar. 2021, doi: 10.1016/j.patrec.2021.03.016.

[16] J. Chen and P. S. Yu, "A domain adaptive density clustering algorithm for data with varying density distribution," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 6, pp. 2310–2321, Jun. 2021, doi: 10.1109/TKDE.2019.2954133.

[17] B. Gu, Z. Zhai, C. Deng, and H. Huang, "Efficient active learning by querying discriminative and representative samples and fully exploiting unlabeled data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 9, pp. 4111–4122, Sep. 2021, doi: 10.1109/TNNLS.2020.3016928.

[18] Z. Wang and J. Ye, "Querying discriminative and representative samples for batch mode active learning," presented at the ACM Trans. Knowl. Discovery Data, Chicago, IL, USA, Aug. 2013.

[19] D. Chen, Q. Yang, J. Liu, and Z. Zeng, "Selective prototype-based learning on concept-drifting data streams," *Inf. Sci.*, vol. 516, pp. 20–32, Apr. 2020, doi: 10.1016/j.ins.2019.12.046.

[20] J. Gan and Y. Tao, "Dynamic density based clustering," in *Proc. ACM Int. Conf. Manage. Data*, Chilcago, IL, USA, May 2017, pp. 1493–1507.

[21] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Semisupervised SVM batch mode active learning with applications to image retrieval," *ACM Trans. Inf. Syst.*, vol. 27, no. 3, pp. 1–29, May 2009, doi: 10.1145/1508850.1508854.

[22] Y. Guo and D. E. Schuurmans, "Discriminative batch mode active learning," presented at Proc. 20th Int. Conf. Neural Inf. Process. Syst., New York, NY, USA, Dec. 2007.

[23] O. M. Aodha, N. D. F. Campbell, J. Kautz, and G. J. Brostow, "Hierarchical subquery evaluation for active learning on a graph," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 564–571.

[24] S. J. Huang, R. Jin, and Z. H. Zhou, "Active learning by querying informative and representative examples," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 1936–1949, Oct. 2014, doi: 10.1109/TPAMI.2014.2307881.

[25] D. Ienco, A. Bifet, I. Žliobait, and B. Pfahringer, "Clustering based active learning for evolving data streams," presented at 16th Int. Conf., Discovery Sci., Singapore, Oct. 2013, pp. 79–93.

[26] X. Gui, X. Lu, and G. Yu, "Cost-effective batch-mode multi-label active learning," *Neurocomputing*, vol. 463, pp. 355–367, Feb. 2021, doi: 10.1016/j.neucom.2021.08.063.

[27] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discov. Data Mining*, Portland, OR, USA, Aug. 1996, pp. 226–231.

[28] D. Cheng, Q. Zhu, J. Huang, Q. Wu, and L. Yang, "Clustering with local density peaks-based minimum spanning tree," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 2, pp. 374–387, Jul. 2019, doi: 10.1109/TKDE.2019.2930056.

[29] Z. Bian, F.-L. Chung, and S. Wang, "Fuzzy density peaks clustering," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 7, pp. 1725–1738, Jul. 2021, doi: 10.1109/TFUZZ.2020.2985004.

[30] T. Qiu and Y. Li, "Fast LDP-MST: An efficient density-peak-based clustering method for large-size datasets," *IEEE Trans. Knowl. Data Eng.*, early access, Feb. 11, 2022, doi: 10.1109/TKDE.2022.3150403. [Online]. Available: https://ieeexplore.ieee.org/document/9712197

[31] M. Du, S. Ding, and H. Jia, "Study on density peaks clustering based on k-nearest neighbors and principal component analysis," *Knowl.-Based Syst.*, vol. 99, pp. 135–145, May 2016, doi: 10.1016/j.knosys.2016.02.001.

[32] T. Kamali and D. W. Stashuk, "Discovering density-based clustering structures using neighborhood distance entropy consistency," *IEEE Trans. Computat. Social Syst.*, vol. 7, no. 4, pp. 1069–1080, Aug. 2020, doi: 10.1109/TCSS.2020.3003538.

[33] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, "MOA: Massiveonline analysis," *J. Mach. Learn. Res.*, vol. 11, pp. 1601–1604, May 2010.

[34] S. H. Bach and M. A. Maloof, "Paired learners for concept drift," in *Proc. 8th IEEE Int. Conf. Data Mining*, Dec. 2008, pp. 15–19.

[35] G. Jaber, C. Antoine, and P. Tarroux, "A new on-line learning method for coping with recurring concepts: The ADACC system," presented at the Int. Conf. Neural Inf. Process., Venue EXCO, Daegu, South Korea, Nov. 2013, pp. 595–604.

[36] Y. M. Wen and S. Liu, "Semi-supervised classification of data streams by BIRCH ensemble and local structure mapping," *J. Comput. Sci. Technol.*, vol. 35, no. 2, pp. 295–304, Mar. 2020.

[37] N. Cesa-Bianchi, C. Gentile, and L. Zaniboni, "Worst-case analysis of selective sampling for linear classification," *J. Mach. Learn. Res.*, vol. 7, pp. 1205–1230, Dec. 2006, doi: 10.1007/s10450-006-0008-8.

KUANGYAN ZHANG was born in Hefei, Anhui, China, in 1997. He received the B.E. degree in engineering from Anhui Polytechnic University, Wuhu, China, in 2020, where he is currently pursuing the M.E. degree with the School of Computer and Information.

His current research interests include online machine learning and knowledge engineering.

SANMIN LIU received the Ph.D. degree in computer science from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2015.

He was a Visiting Associate Professor with the Department of Computing, Macquarie University, Sydney, NSW, Australia, from May 2019 to March 2020. He is currently a Professor with the School of Computer and Information, Anhui Polytechnic University, Wuhu, China. His current research interests include data mining and machine learning.

YANFEI CHEN received the B.E. degree in software engineering from Shanghai Polytechnic University, Shanghai, China, in 2020. She is currently pursuing the M.E. degree in software engineering with Anhui Polytechnic University, Anhui, China.

Her research interests include machine learning and data mining.