Applied Intelligence (2022) 52:13250–13267
https://doi.org/10.1007/s10489-021-03139-y

A clustering-based active learning method to query informative and representative samples

Xuyang Yan[1] · Shabnam Nazmi[1] · Biniam Gebru[1] · Mohd Anwar[1] · Abdollah Homaifar[1] · Mrinmoy Sarkar[1] · Kishor Datta Gupta[1]

Accepted: 13 December 2021 / Published online: 25 February 2022

Abstract

Active learning (AL) has widely been used to address the shortage of labeled datasets. Yet, most AL techniques require an initial set of labeled data as the knowledge base to perform active querying. The informativeness of the initial labeled set significantly affects the subsequent active query; hence the performance of active learning. In this paper, a new clustering-based active learning framework, namely Active Learning using a Clustering-based Sampling (ALCS), is proposed to simultaneously consider the representativeness and informativeness of samples using no prior label information. A density-based clustering approach is employed to explore the cluster structure from the data without requiring exhaustive parameter tuning. A simple yet effective distance-based querying strategy is adopted to adjust the sampling weight between the center-based and boundary-based selections for active learning. A novel bi-cluster boundary-based sample query procedure is introduced to select the most uncertain samples across the boundary among adjacent clusters. Additionally, we developed an effective diversity exploration strategy to address the redundancy among queried samples. Our extensive experimentation provided a comparison of the ALCS approach with state-of-the-art methods, exhibiting that ALCS produces statistically better or comparable performance than state-of-the-art methods.

Keywords Active learning · Clustering · Informative-based query · Representative-based query · Center-based selection · Boundary-based selection

1 Introduction

With the exponential growth of data, the scarcity of associated labels has challenged passive learning approaches for predictive modeling. As a solution, various active learning (AL) [23] approaches have been proposed for real-world applications such as image annotations [43, 50], text classifications [35, 39], and speech recognition [63]. AL techniques are capable of interactively querying human experts for labels of a small number of representative samples and building an accurate predictive model using those labels.

Generally, there are two types of query strategies in AL techniques [12, 17]: (i) *informativeness-based query (IBQ)*; and (ii) *representativeness-based query (RBQ)*. The *IBQ* evaluates the amount of information from samples with respect to a statistical model and selects samples that help to reduce the uncertainty of that model. The use of statistical models in *IBQ* necessitates prior label information to train such models. Therefore, the efficacy of these approaches primarily depends on the quality of the initial label set. Moreover, *IBQ* approaches are more likely to select samples that are close to the decision boundary which leads to sampling bias [17] in AL. To address the challenges with *IBQ*, clustering is widely used to explore the *representativeness* of samples in AL. In clustering-based AL approaches, samples are assumed to share the same class label within the same cluster so that *RBQ* is conducted by querying a single instance from each cluster [7]. Recently, some studies in [17, 44, 45, 47] proposed a hybrid framework of *IBQ* and *RBQ* for clustering-based AL learning through a combination of the center-based and the boundary-based selection methods. Clustering-based AL methods have shown promising performance in the literature [7, 17, 44, 45, 47].

Despite the substantial success of the existing clustering-based AL methods, several challenges remain. First of all,

✉ Abdollah Homaifar
homaifar@ncat.edu

Extended author information available on the last page of the article.

the performance of existing clustering-based AL methods strongly depends on the clustering accuracy [17], and the optimization of clustering parameters have a direct influence on the clustering quality. To the best of authors' knowledge, no prior work has been conducted on employing a clustering procedure that does not require parameter optimization in AL problems. Secondly, the effect of sampling weight between *IBQ* and *RBQ* is rarely investigated in clustering-based AL methods. A more flexible hybrid sample selection method should be developed to adjust the weights of *IBQ* and *RBQ* for AL with respect to the distributions of clusters [24, 25]. Thirdly, the existing boundary-based selection strategy primarily queries labels for the farthest samples in each cluster without considering neighboring clusters [45, 47]. Since samples that are on the boundary region of adjacent clusters have higher classification uncertainty, it is necessary to conduct the label query from the cross-boundary region. Finally, the redundancy among queried samples is extensively investigated for *IBQ* methods [18, 20, 48, 51, 53] while limited efforts [47, 51] are made on clustering-based AL methods to consider the diversity among queried samples.

In this paper, a clustering-based AL method, namely **AL** utilizing a **C**lustering-based **S**ampling (ALCS), is proposed to actively select both informative and representative samples using no prior label information. Considering the limitations of the existing clustering-based AL approaches, ALCS adopt a density-based clustering technique, namely fitness proportionate sharing clustering (FPS-clustering) [54], to relax the dependency on clustering parameter optimization. We developed a new hybrid sample selection strategy for ALCS framework to consider the effect of sampling weight between *IBQ* and *RBQ* with respect to the density of clusters. Furthermore, we propose a bi-cluster boundary-based selection procedure to improve the performance of the new hybrid selection strategy. Additionally, an effective diversity exploration strategy is introduced to reduce the redundancy among active queried samples.

In summary, the major contributions of this manuscript are as follows:

- Developed a clustering-based AL framework to address the dependency of state-of-the-art AL techniques on the initial label set. The proposed framework utilizes an effective density-based clustering technique without requiring parameter optimization.
- Employed a novel hybrid method of center-based and boundary-based sample selection to adjust the sampling weight between *IBQ* and *RBQ* based on the density of each cluster. The one-sigma principle is used to determine the sampling weights in each cluster.
- Proposed a bi-cluster boundary-based selection procedure to select informative samples from the cross-boundary regions of the neighboring clusters. Mathematical justifications are provided to support the bi-cluster boundary-based selection strategy.
- Developed an effective diversity exploration technique (EDET) to address the redundancy among queried samples. Experimental results and statistical analysis are presented to justify its efficacy.

The remainder of this paper is organized as follows: Section 2 reviews the state-of-the-art AL approaches. Section 3 provides the background of this research including basic definitions and the FPS-clustering procedure. The details of ALCS are discussed in Section 4, and Section 5 presents the experimental results. A comparison between the proposed approach and several other state-of-the-art methods is carried out and discussed in Section 5. Section 6 concludes the paper and outlines future works.

2 Related work

As mentioned previously, the existing AL approaches can be categorized into *IBQ* and *RBQ*. Most *IBQ* methods primarily focus on querying labels for samples with higher classification uncertainty and thus require a well-trained initial model. In [2, 22, 39, 51, 61], the uncertainty-based sampling (UBS) is used to evaluate the informativeness of samples using traditional classifiers such as Support Vector Machine (SVM), Decision Tree (DT), and Naive Bayes (NB) classifiers. Then, samples with minimal classification confidence are considered as informative samples. Later, the Extreme Learning Machine (ELM) classifier [16] is used as the base classifier to perform the UBS in [64]. The error-correcting output codes are utilized to measure the uncertainty of samples for multi-class AL problems [12]. Most recently, a **DU**al **A**ctive **L**earning (DUAL) for both model and data selection [38] was developed to mitigate the possible drawbacks when the quality of classification model is poor.

Considering the limitation of utilizing a single classifier, query-by-committee (QBC) [6, 11, 20, 36, 37] methods establish a committee of classifiers using the initially available labeled information and select samples which caused the most conflicts among the committee members. Also, several other *IBQ* AL approaches, including expected error reduction [15, 33], expected model change [19, 35], variance reduction [13, 34] and "Min-max" view active learning [14, 59], were proposed to actively select informative samples. As an extension of "Min-max" view active learning methods, the maximum variance for active learning (MVAL) [60] queried both the informative and representative instances. In [21], a cost-sensitive AL method was developed to address the different labeling costs for different classes using the combination of UBS and expected error reduction strategies. An online AL framework, termed Passive-Aggressive Active (PAA) [26], was introduced to improve the learning performance using both the misclassified samples and correctly classified samples. Several diversity-based AL methods were employed to address the redundancy among informative samples for UBS and QBC methods in [18, 20, 48, 51, 53].

As an alternative to *IBQ* approaches, *RBQ* AL methods were proposed to query labels for a subset of representative samples that reflects the distributions of the unlabeled data. In [3, 4, 41, 49], a number of Maximum Mean Discrepancy (MMD) AL methods are developed to explore the representativeness of samples. Clustering-based AL approach is another group of *RBQ* approaches that were extensively investigated. In [7], a hierarchical clustering approach was used to partition data into a set of clusters and the centers of clusters are selected as the representative samples for label querying. A cost effective AL framework is developed to simultaneously minimize the labeling costs and classification errors using hierarchical clustering [40]. In [10] and [30], the k-medoids clustering procedure was utilized to explore the data structure and identify the centers as samples with high *representativeness*. Wang and Fan [44] proposed active learning through density clustering (ALEC) by utilizing a density peak clustering procedure [32]. In [45], a two-stage clustering procedure was proposed to reduce the high computational complexity of ALEC.

A formal framework, namely active learning by **QU**erying **I**nformative and **R**epresentative **E**xamples (QUIRE) [17], was developed to combine the *IBQ* and *RBQ* together in AL. In [42], the authors introduced a pre-clustering procedure to explore the representativeness of samples and simultaneously consider the informativeness of samples using the “close-to-boundary” criteria. Recently, the hybrid of center-based and boundary-based selections was adopted to perform the *IBQ* and *RBQ* for clustering-based AL approaches in [27, 29, 44, 47, 51]. In [27], the authors used the center-based selection to train the initial classification model and employed the boundary-based selection for the subsequent refinement of the model. Unlike [27], the center-based and boundary-based selections were performed simultaneously for active label querying in [29, 44, 47]. In [29] and [47], efforts were also employed to enhance the clustering performance for subsequent active label querying procedure. To address the redundancy among queried samples, a multi-standard active learning (MSAL) [51] strategy was discussed to integrate the diversity criteria with the hybrid sample selection.

In the proposed AL framework, we employed a new hybrid sample selection strategy to enable the adjustment of sampling weights between *IBQ* and *RBQ* based on the clustering distributions. Also, a bi-cluster boundary-based selection policy is introduced to enhance the learning performance of *IBQ*. Instead of modifying the active label querying criteria, a diversity exploration procedure is incorporated into the proposed hybrid sample selection to address the redundancy among queried samples.

3 Preliminaries

In this section, some basic definitions about AL are provided and a brief review of the FPS-clustering method is discussed.

3.1 Basic definitions

Let the unlabeled and labeled datasets be denoted as X_U and X_L, respectively, such that $X_U = \{\mathbf{x}_i^U | \mathbf{x}_i^U \in \mathcal{R}^m, i = 1, \ldots, n_U\}$, $X_L = \{(\mathbf{x}_i^L, y_i) | \mathbf{x}_i^L \in \mathcal{R}^m, y_i \in Y, i = 1, \ldots n_L\}$, and $|Y| = n_c$. Moreover, Q_U is the set of representative samples selected from the unlabeled dataset with respect to different criteria where $|Q_U| = n_q$. We use C_i to represent the i^{th} cluster center extracted from X_U and $\mathbf{d}(C_i)$ refers to the inner-cluster distance matrix in cluster i. These notations are used throughout the paper. The definitions of the *IBQ* and *RBQ* criterion can be expressed as follows:

Definition 1 (Informativeness-based query) Let the true class probability distribution be $P(y|X_U \cup X_L)$ and the estimated class probability distribution be $\hat{P}(y|X_L \cup Q_U)$. This query method, selects a set of informative samples Q_U with respect to the following objective:

$$\min L(P(y|X_U \cup X_L) - \hat{P}(y|X_L \cup Q_U)). \tag{1}$$

Here, $L(\cdot)$ denotes the loss function.

Definition 2 (Representativeness-based query) Assume X_U follows the distribution $p(X_U)$. This query method, selects a set of representative samples Q_U with respect to the following objective:

$$\min L(p(X_U) - \hat{p}(Q_U)). \tag{2}$$

In this equation, $\hat{p}(Q_U)$ is the estimated data distribution from the representative samples Q_U.

These definitions are used in this paper to develop the sample query strategies with respect to both criteria.

3.2 FPS-clustering

To address the dependency of density-based clustering approaches on certain clustering parameters, FPS-clustering is proposed to address exhaustive parameter tuning in [54–57]. Hence, we employ the FPS-clustering to address the limitation of clustering-based AL approaches in terms of parameter optimization. FPS-clustering transforms the clustering problem into the problem of searching for multiple density peaks. A Gaussian kernel function is used to model the density distribution of the dataset and it is defined as follows:

$$\mathcal{F}(\mathbf{x}_i) = \sum_{j=1}^{n_U} (e^{-\frac{D(i,j)^2}{\beta}})^{\gamma}. \tag{3}$$

Here, D is the Euclidean-distance matrix which consists of the pairwise-distances between unlabeled instances in X_U and n_U denotes the total number of unlabeled samples in the dataset. The distance between the samples $\mathbf{x}_i$ and $\mathbf{x}_j$ is denoted by $D(i, j)$. The normalization parameter β is approximated by the variance of the dataset. The stabilization parameter γ controls the shape of the cluster and can be estimated through the correlation comparison algorithm (CCA) [58]. In the FPS-clustering framework, the density values for individual points serve as a fitness measure for evaluating their potential to become a cluster center. Algorithm 1 summarizes the general procedure of the FPS-clustering method.

Algorithm 1 FPS-clustering.

```
Input: $X_U$
Parameter: the list of the temporary potential centers TPC
Output: Cluster information $\Omega$
Estimate the parameters $\beta$ and $\gamma$ using the variance of $X_U$ and CCA
Evaluate fitness of each sample using (3)
$TPC = \emptyset$
while Not all samples are clustered do
    Rank fitness in descending order and insert the highest-ranked sample into TPC
    Explore the neighborhood of the current TPC to identify samples that belong to it and marked them as clustered
    Update fitness values of samples in the neighborhood of the current highest ranked sample through FPS
end while
Perform the merge among TPCs by checking the existence of density “valley” until no members from TPC could be merged
Obtain cluster information in terms of cluster center ($C_i$) and inner-cluster distance vector ($\mathbf{d}(C_i)$)): $\Omega = \{(C_i, \mathbf{d}(C_i))|i = 1, ..., c\}$
return $\Omega$
```

In Algorithm 1, the cluster analysis starts with the fitness evaluation for each point in the dataset, and a temporary potential center (*TPC*) is identified as the point with the highest fitness. The fitness-proportionate sharing (FPS) procedure is applied to the samples which lie in the neighborhood of the *TPC* and it scales down the fitness values of samples that are close to the identified *TPC*, which reduces the chance for those samples to become the next *TPC*. This procedure can effectively guide the search for *TPCs* by avoiding unnecessary explorations for *TPCs*. To eliminate highly overlapping *TPCs*, a subsequent expansion procedure is employed by locating two neighboring *TPCs* with an existing “*valley*” [54] between them and merging them into a single cluster. A “*valley*” exists when there is a drop in the density distribution function between the two peaks associated with the neighboring *TPCs*.

4 Proposed ALCS technique

In this section, the proposed ALCS technique is discussed in terms of its two main components: (i) clustering; and (ii) distance-based instance selection. The workflow of the ALCS technique is presented in Fig. 1. ALCS utilized the FPS-clustering to explore the structure of data to relax the clustering parameter optimization. Then, a new hybrid sample selection procedure with the bi-cluster boundary-based selection is employed to consider the sampling weights as well as diversity among queried samples. Additionally, the time complexity of the proposed ALCS method is provided in this section.

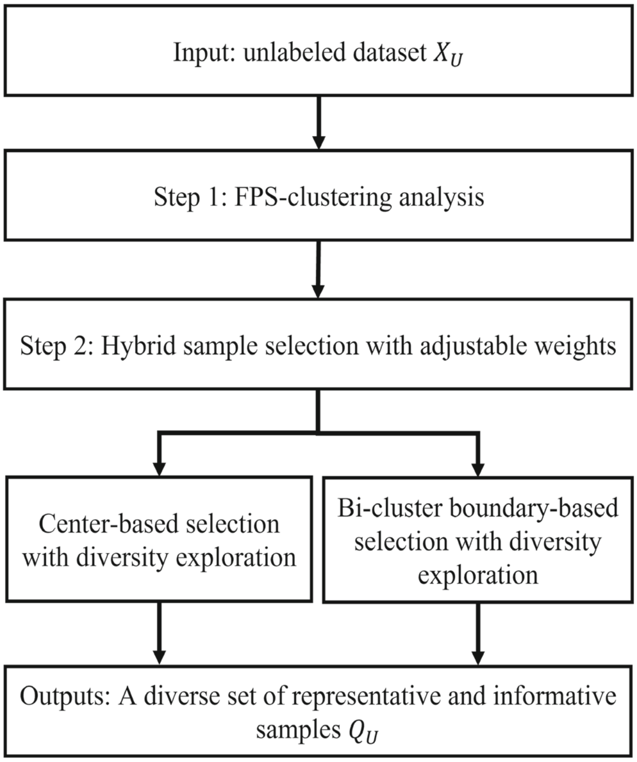

Input: unlabeled dataset X_U
Step 1: FPS-clustering analysis
Step 2: Hybrid sample selection with adjustable weights
Center-based selection with diversity exploration
Bi-cluster boundary-based selection with diversity exploration
Outputs: A diverse set of representative and informative samples Q_U

Fig. 1 A workflow of the proposed clustering-based AL framework

4.1 Clustering

To effectively alleviate the exhaustive parameter tuning issue, the FPS-clustering algorithm is employed to discover the cluster information as the first step of the ALCS technique. The FPS-clustering algorithm takes the unlabeled dataset X_U as the input and then outputs a set of clusters and the corresponding cluster information Ω, which is expressed as follows:

$$\Omega = \{(C_i, \mathbf{d}(C_i))|i = 1, ..., c\}, \tag{4}$$

$$\mathbf{d}(C_i) = \{d(\mathbf{x}_i^j, C_i)|j = 1, ..., |C_i|\}. \tag{5}$$

Here, C_i refers to the center of the i^{th} cluster, c denotes the total number of discovered clusters, and $d(\mathbf{x}_i^j, C_i)$ is the distance from sample $\mathbf{x}_i^j$ to its respective cluster center C_i. The cardinality $|C_i|$ denotes the number of samples that belong to C_i.

4.2 Distance-based sample selection

The hybrid of center-based and boundary-based sample selections has been proposed in [27, 45, 47] to satisfy the *IBQ* and *RBQ* criteria for clustering-based AL methods. In this paper, we developed a new hybrid sample selection process by: (i) enabling the adjustment of sampling weight between the *IBQ* and *RBQ* with respect to the density of each cluster; (ii) introducing a novel bi-cluster boundary-based selection procedure to select the most informative samples using the law of cosines; and (iii) employing an effective diversity exploration technique for queried samples. The details are discussed below.

4.2.1 Hybrid sample selection strategy

Let the number of queried samples from the i_{th} cluster be n_{q_i}. The bi-cluster boundary-based selection step takes $\lfloor n_{q_i} \times \rho_i \rceil$ samples from cluster i as $Q_{boundary}$ where ρ_i denotes the sampling weight from the boundary of two adjacent clusters. Accordingly, the center-based selection policy chooses the remaining $\lfloor n_{q_i} \times (1 - \rho_i) \rceil$ samples from the center region as Q_{center}. The value of ρ_i ranges from zero to one and we employ an effective procedure to select a proper ρ_i based on the density of each cluster.

Assume that the radius of the cluster i is R_i. The mean and standard deviation of all cluster radius are denoted as μ_R and σ_R, respectively. Based on the cluster radius, ALCS partitions all clusters into two groups: *dense cluster* and *sparse cluster*. The cluster i is considered as a *dense cluster* if $R_i \leq |\mu_R - \sigma_R|$; otherwise, the i_{th} cluster is a *sparse cluster*. This idea is motivated via the one-sigma principle from Gaussian distribution to identify clusters that have significantly small radius with a confidence level of 66.67%. For a *dense cluster*, the samples are more likely to share similar characteristics. Thus, the value of ρ_i is chosen as zero to conduct *RBQ* only. In a *sparse cluster*, samples are relatively far from each other and the uncertainty of samples becomes relatively high. Consequently, we set the value of ρ_i to be 0.5 such that samples are considered for *IBQ* and *RBQ* with equal importance. During center-based and boundary-based selection stage, EDET is applied to improve the quality of queried samples from the diversity perspective. A general algorithmic description of the new hybrid sample selection procedure is summarized in Algorithm 2.

Algorithm 2 The hybrid sample selection strategy.

```
Input: $\Omega, n_q, \rho_i$
Parameter: $Q_{center}$, $Q_{boundary}$, and $n_{q_i}$
Output: The set of queried samples $Q_U$
$Q_U = \emptyset$
for i = 1 to $n_C$ do
    Calculate the number of queries for cluster $i$: $n_{q_i} = \lfloor \frac{|C_i|}{n_U} \times n_q \rceil$
    Perform the center-based query to obtain $Q_{center}$ using Algorithm 3 and conduct the diversity exploration
    Perform the bi-cluster boundary-based query to obtain $Q_{boundary}$ using Algorithm 4 and conduct the diversity exploration
    $Q_U = Q_U \cup \{Q_{center} \cup Q_{boundary}\}$
end for
return $Q_U$
```

4.2.2 Center-based selection

In clustering-based AL approaches, the center-based selection policy is widely used to choose the most representative samples from each cluster. Nonetheless, the existing clustering-based AL methods ignore the effect of sampling weight and simply apply the center-based selection [28, 29, 44, 47, 51]. Hence, the proposed hybrid selection procedure aims to improve the learning performance by adjusting the sampling weight of the center-based selection according to the density of the cluster. To perform the center-based selection, the query priority of each sample is computed in terms of the **C**luster **R**epresentativeness (CR). Let $CR(*)$ and $P(*)$ be the cluster representativeness and query priority functions for clustered samples, respectively. For center-based selection, the query priority of x_i^j is calculated below.

$$P(x_i^j) = CR(x_i^j), \tag{6}$$

and

$$CR(x_i^j) = \frac{1}{1 + e^{d(x_i^j, C_i)}}. \tag{7}$$

Where $d(x_i^j, C_i)$ refers to the distance from x_i^j to C_i. From (7), the representativeness of each sample is inversely proportional to its distance to C_i and samples that are close to the cluster center have higher representativeness.

Accordingly, the implementation of the center-based selection procedure is presented in Algorithm 3.

Algorithm 3 Center-based sample selection.

```
Input: $\Omega$, $n_{q_i}$, $\rho_i$
Parameter: $\mathbf{d}(C_i)$
Output: $Q_{center}$
Calculate the representativeness of samples using (6) and (7)
for $j = 1 : \lfloor n_{q_i} \times (1 - \rho_i) \rceil$ do
    Sort samples in the $i^{th}$ cluster in descending order based on $CR$
    Insert the first sample from the sorted list into $Q_{center}$
    Apply the diversity exploration strategy in Section 4.3
end for
return $Q_{center}$
```

4.2.3 Bi-cluster boundary-based selection

Considering the label scarcity, the conventional boundary-based selection strategy assumes that samples which are closer to the cluster boundary have higher classification uncertainty and conducts the *IBQ* using this assumption. In [44, 47], the authors considered the distance from each sample to its assigned cluster centers to choose the farthest sample as the most uncertain sample. Since samples from the cross-boundary regions of neighboring clusters have higher classification uncertainty, it is more valuable to query labels for those samples. Hence, we propose an effective bi-clusters boundary-based selection strategy to identify the most uncertain samples using the distance to their assigned cluster center and neighboring cluster center. This strategy utilizes the law of cosines to query the most informative samples from the cross-boundary region with two adjacent cluster centers. Specifically, each cluster provides a set of candidates for the bi-boundary samples by selecting $\frac{|C_i|}{2}$ samples with the largest distance to the cluster center. Then, the selection of the bi-boundary samples can be performed by choosing candidates that are close to the cross-boundary with two neighboring clusters.

Assume the candidate bi-boundary sample in the i^{th} cluster is $x^j_{CB_i}$ and the candidate set is $CB = \{x^j_{CB_i} | j = 1, ..., \frac{|C_i|}{2}\}$. The two adjacent cluster centers are denoted as NC_1 and NC_2. The query priority of $x^j_{CB_i}$ is calculated using the **C**luster **U**ncertainty (CU), which is expressed as follows:

$$P(x^j_{CB_i}) = CU(x^j_{CB_i}), \tag{8}$$

and

$$CU(x^j_{CB_i}) = \frac{1}{1 + e^{\frac{d_1+d_2}{d_{ref_1}+d_{ref_2}}}}. \tag{9}$$

Where d_{ref_1} and d_{ref_2} denotes the distance from C_i to its two neighboring cluster centers. Here, d_1 refers to the distance from $x^j_{CB_i}$ to NC_1 and d_2 refers to the distance from $x^j_{CB_i}$ to NC_2, respectively. According to (9), the following theorem can be defined.

Theorem 1 *A candidate bi-boundary sample $x^j_{CB_i}$ is considered to have higher uncertainty when it has a larger value of CU. Conversely, a smaller value of CU indicates $x^j_{CB_i}$ has lower uncertainty.*

Proof Let the distance between two neighboring cluster centers of C_i be $d_{NC_{1,2}}$ and the angle between d_1 and d_2 be θ, the following equation can be obtained using the triangle principle.

$$d_1^2 + d_2^2 - 2d_1d_2cos(\theta) = d^2_{NC_{1,2}}. \tag{10}$$

Based on (10), the variation is derived below:

$$(d_1 + d_2)^2 = d^2_{NC_{1,2}} + 2d_1d_2 \times (1 + cos(\theta)). \tag{11}$$

Assume the area of the triangle $\{x^j_{CB_i}NC_1NC_2\}$ is $A(x^j_{CB_i})$ such that $A(x^j_{CB_i}) = \frac{1}{2}d_1d_2sin(\theta)$, (11) can be rewritten as follows:

$$(d_1 + d_2)^2 = d^2_{NC_{1,2}} + 4 \times A(x^j_{CB_i}) \times \frac{(1 + cos(\theta))}{sin(\theta)}. \tag{12}$$

Since $sin(\theta) = 2sin(\frac{\theta}{2})cos(\frac{\theta}{2})$ and $1+cos(\theta) = 2cos^2(\frac{\theta}{2})$, we can obtain the following equation:

$$(d_1 + d_2)^2 = d^2_{NC_{1,2}} + 4 \times A(x^j_{CB_i}) \times \frac{cos(\frac{\theta}{2})}{sin(\frac{\theta}{2})}. \tag{13}$$

Finally, (11) can be expressed as:

$$(d_1 + d_2)^2 = d^2_{NC_{1,2}} + 4 \times A(x^j_{CB_i}) \times cot(\frac{\theta}{2}). \tag{14}$$

As shown in Fig. 2, the candidate bi-boundary sample $x^{j_1}_{CB_i}$ is much closer to the cross-boundary region than $x^{j_2}_{CB_i}$ such that x^{j_1} have a larger θ and a smaller $A(x^j_{CB_i})$ than x^{j_2}. According to (14), the sum of d_1 and d_2 monotonically decreases if $A(x^j_{CB_i})$ decreases and $\frac{\theta}{2}$ increases. This property indicates that candidates should have smaller sum of d_1 and d_2 if they are close to the cross-boundary region. Since the sum of d_{ref_1} and d_{ref_2} is fixed, candidates from the cross-boundary region will have higher value of CU. □

Based on Theorem 1, a bi-cluster boundary-based selection procedure is proposed and Algorithm 4 outlines the proposed boundary-based selection procedure.

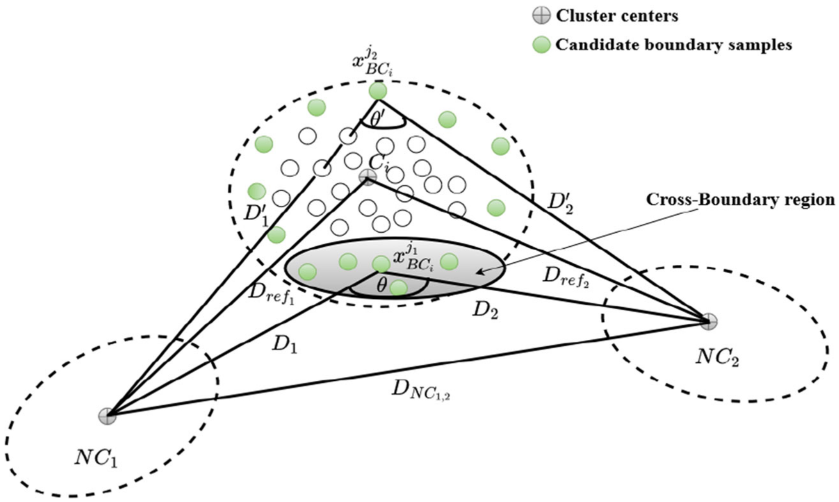


Fig. 2 An illustrative example with two types of candidate bi-boundary samples from cluster i in two-dimensional space. The candidate $x^1_{CB_i}$ is located in the cross-boundary region between cluster i and its two nearest clusters (NC_1 and NC_2). The other candidate boundary sample $x^2_{CB_i}$ is far away from the cross-boundary region

Algorithm 4 Bi-cluster boundary-based sample selection.

Input: Ω, n_{q_i}, ρ_i
Parameter: a set of candidates for bi-cluster boundary samples $Q_{boundary_C}$
Output: $Q_{boundary}$

```
$CB = \emptyset$
Sort samples in the $i^{th}$ cluster in ascending order based on $\mathbf{d}(C_i)$
Insert the last $\frac{|C_i|}{2}$ samples from the sorted list into $CB$
for $j = 1 : \lfloor n_{q_i} \times \rho_i \rceil$ do
    Sort samples in $CB$ in ascending order using (8) and (9)
    Insert the first ranked samples in $CB$ into $Q_{boundary}$
    Apply the diversity exploration strategy for $CB$ in Section 4.3
end for
return $Q_{boundary}$
```

4.3 Diversity exploration for active label querying

Diversity [47] is another well-known challenge that accounts for the redundancy among queried samples in AL problems. Several recent AL approaches [20, 46, 48, 53], directly incorporate diversity into the evaluation metric to handle the redundancy among informative samples. Alternatively, we developed a diversity exploration strategy based on Fitness proportionate niching (FPN) [52] to guide the search of informative and representative samples. FPN was initially proposed to maintain the diversity in the population set-based genetic algorithm and has shown substantial success in multi-objective optimization problems [31]. Let X_{C_i} be a set of samples that belongs to C_i, the query priority function is expressed as follows.

$$P(X_{C_i}) = \begin{cases} CR(X_{C_i}), & \text{query from centers;} \\ CU(X_{C_i}), & \text{query from boundaries.} \end{cases} \tag{15}$$

Where CR and CU denote cluster representativeness and cluster uncertainty, respectively. EDET aims to decompose X_{C_i} into a number of small niches and query a set of diverse samples from different niches. During the center-based selection procedure, the center sample has the highest query priority and is inserted into the queried sample set initially. Then, a niche can be formed by a set of samples in the neighborhood of the center sample and a priority sharing strategy is employed to decrease the query priorities of other samples in the niche. The average distance for all k-nearest-neighbor graphs within a cluster is used as the neighborhood radius. As a rule of thumb, we set the value of k to be the square root of cluster size. Assume n^j_i and $X_{n^j_i}$ denote the j^{th} niche in C_i and a set of samples belong to n^j_i, respectively. Equation 16 describes the priority sharing function.

$$P(X_{n^j_i}) = \frac{P(X_{n^j_i})}{\sum P(X_{n^j_i})}, x_i \in n^j_i. \tag{16}$$

From (16), samples from the same niche will have relatively low priorities during the next sample query stage. Consequently, it guarantees to query more diverse samples from other niches that locate in the center regions of clusters. This procedure repeats until $\lfloor n_{q_i} \times (1 - \rho_i) \rceil$ samples are queried and finally, a diverse set of representatives is queried from different parts of the cluster centers. Figure 3 displays a visualization of EDET for center-based selection using (16). Similarly, the proposed diversity exploration technique is employed for the bi-cluster boundary-based selection strategy to improve the diversity among queried boundary samples.

4.4 Time complexity

As discussed previously, ALCS consists of two steps and the time complexity analysis of each step is outlined directly. Considering that the calculations of distances among samples take the most computational power, we quantify the overall time complexity by means of the total

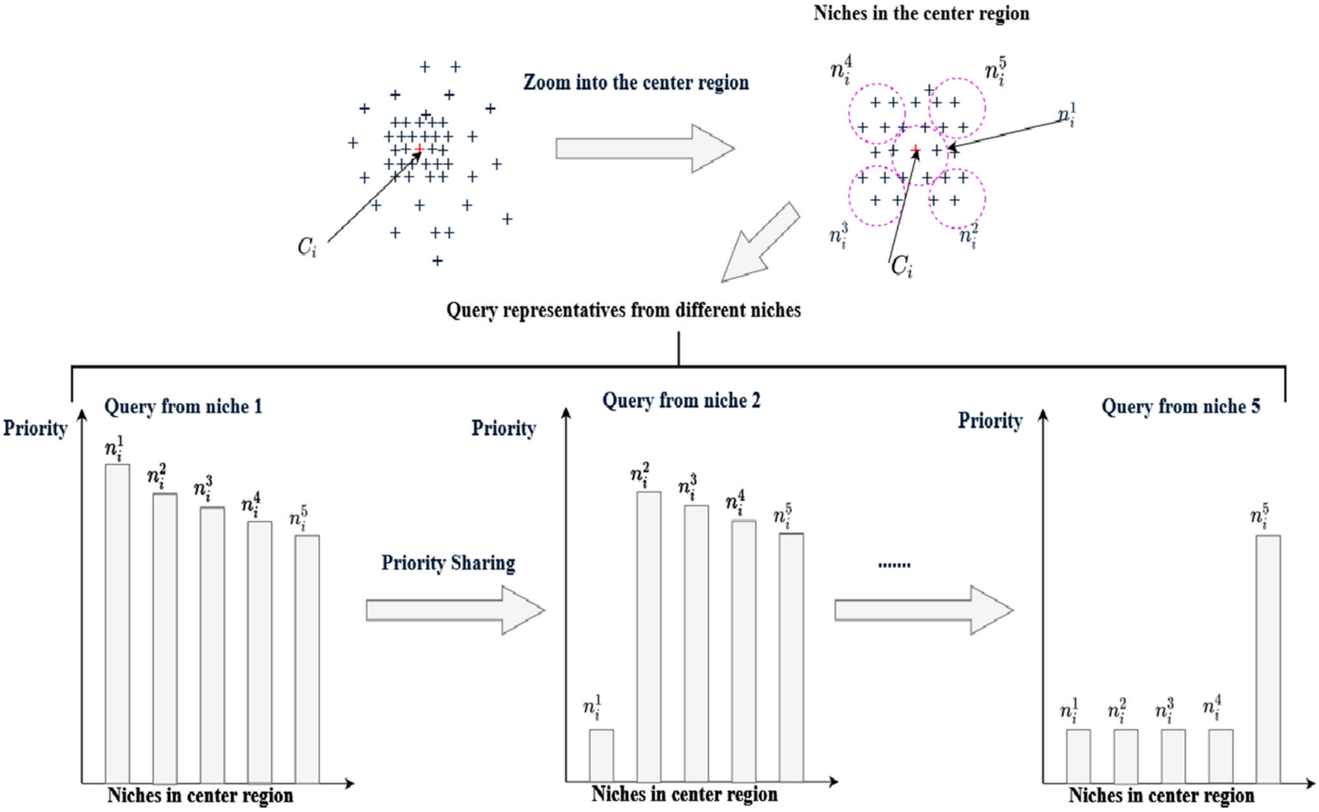


Fig. 3 An illustrative example of the proposed diversity exploration strategy for center-based selection. Here, the center region is partitioned into five small niches. The center-based query starts by querying a representative from n_i^1 and then perform the priority sharing using (16). After sharing, all samples in n_i^1 will have relatively small query priorities, which enable the next sample query from n_i^2. This procedure repeats until all niches have been considered for querying

number of distance calculations. Assume there are n_U unlabeled samples and each sample has m features, the time complexity of each step in terms of the number of distance calculations is provided as follows:

- Step 1: In cluster analysis, it requires $O(n_U^2 m)$ distance calculations;
- Step 2: In distance-based querying stage, it requires $O(1)$ distance calculations;

Therefore, the overall time complexity of the ALCS framework can be expressed as: $O(n_U^2 m) + O(1) = O(n_U^2 m)$.

5 Experiments and results discussion

In this section, experiments are conducted on the proposed framework using twelve benchmark datasets and the results are discussed. We present two comparison studies, including the comparison with classifier-based AL and clustering-based AL approaches, to evaluate the efficacy of ALCS framework. We also used two classification methods for the comparison with classifier-based AL approaches. For these experiments, the following major questions are addressed:

(1) Does the diversity exploration strategy improve the performance of the ALCS framework?
(2) Does ALCS achieve better performance than the existing classifier-based AL approaches?
(3) Does ALCS consistently perform well against classifier-based AL methods on different classification techniques?
(4) Does ALCS provide better performance than the clustering-based AL techniques?

5.1 Datasets

To evaluate the efficacy of the proposed AL approach, twelve benchmark datasets from [9] are used in the experiments and these datasets are widely used in AL research [44, 45]. It includes ten real-world and two synthetic datasets. Table 1 summarizes the properties of these datasets in terms of the number of samples, dimensions, classes, sources, and domains. As shown in Table 1, the dimension of benchmark datasets ranges from

Table 1 Description of the datasets used in the experiments

Datasets	Sample size	Dimensions	Classes	Source	Domain
R15	600	2	15	Synthetic	NA
Australian	690	14	2	Real	Finance
Aggregation	788	2	7	Synthetic	NA
Vehicle	846	18	4	Real	Life
Spambase	4601	57	2	Real	Text
Waveform	5000	40	3	Real	Physical
Electricity	10000	14	2	Real	Life
DLA0.01	10000	17	5	Real	Society
Penbased	10992	16	10	Real	Computer
GasSensor	13910	128	6	Real	Chemical
DCCC	30000	23	2	Real	Finance
MNIST	70000	784	10	Real	Image

2 to 784. Also, the sample size of all twelve datasets varies from 600 to 70000. Four of the experimental datasets, including Aggregation, Penbased, DCCC, and DLA0.0, are imbalanced datasets.

5.2 Compared AL methods

For comparison purposes, two groups of state-of-the-art methods, including four classifier-based and five

Table 2 Performance comparison of the ALCS and the classifier-based AL methods using *k*NN. (The relative rank of each algorithm is shown within the parentheses)

Datasets	Metrics	UBS	QBC	DWUS	MVAL	ALCS_N	ALCS_D
R15	*Acc*	99.21(2)	**99.28(1)**	98.89(5)	96.33(6)	98.93(4)	99.07(3)
	F_{mac}	99.17(1)	**99.11(2)**	98.02(5)	95.78(6)	98.61(4)	99.06(3)
Australian	*Acc*	82.02(3)	81.63(4)	78.47(6)	78.75(5)	82.14(2)	**83.31(1)**
	F_{mac}	82.55(2)	81.40(4)	78.25(5)	78.11(6)	81.99(3)	**83.06(1)**
Aggregation	*Acc*	81.79(5)	85.34(4)	74.60(6)	87.74(3)	90.80(2)	**99.43(1)**
	F_{mac}	62.06(5)	72.81(4)	47.28(6)	77.03(2)	76.51(3)	**99.09(1)**
Vehicle	*Acc*	52.66(5)	**56.14(1)**	49.01(6)	54.65(3)	53.34(4)	54.74(2)
	F_{mac}	52.75(5)	**54.43(1)**	48.83(6)	53.19(3)	52.78(4)	54.30(2)
Spambase	*Acc*	79.45(4)	81.41(2)	78.71(6)	80.36(3)	79.01(5)	**81.54(1)**
	F_{mac}	78.91(3)	80.73(2)	77.66(6)	78.39(5)	78.44(4)	**80.92(1)**
Waveforms	*Acc*	72.04(6)	73.03(5)	74.31(3)	73.64(4)	75.98(2)	**76.66(1)**
	F_{mac}	71.48(6)	72.63(5)	74.14(3)	73.40(4)	75.91(2)	**76.62(1)**
Electricity	*Acc*	81.48(5)	84.60(2)	80.23(6)	82.56(4)	83.40(3)	**85.34(1)**
	F_{mac}	80.75(4)	83.53(2)	76.06(6)	79.58(5)	81.05(3)	**83.76(1)**
DLA0.01	*Acc*	95.19(2)	**96.04(1)**	89.87(6)	91.84(3)	93.46(5)	93.61(4)
	F_{mac}	**95.57(1)**	94.58(2)	88.10(6)	91.79(3)	88.13(5)	88.26(4)
Penbased	*Acc*	89.33(5)	91.91(2)	86.84(6)	89.95(4)	91.44(3)	**94.80(1)**
	F_{mac}	87.64(5)	91.88(2)	84.25(6)	89.14(4)	91.47(3)	**94.76(1)**
GasSensor	*Acc*	67.37(2)	66.33(4)	65.68(6)	66.18(5)	67.03(3)	**72.81(1)**
	F_{mac}	65.11(2)	64.80(3)	60.96(6)	62.35(5)	64.39(4)	**71.55(1)**
DCCC	*Acc*	**76.88(1)**	76.71(4)	76.56(5)	75.98(6)	76.56(2)	76.43(3)
	F_{mac}	56.39(4)	58.85(2)	55.33(6)	57.45(3)	55.41(5)	**60.84(1)**
MINST	*Acc*	91.73(3)	**94.25(1)**	89.42(6)	90.58(5)	90.92(4)	91.83(2)
	F_{mac}	91.55(3)	**94.13(1)**	89.31(6)	89.92(4)	89.75(5)	91.79(2)
Mean ranks	*Acc*	3.58	2.58	5.58	4.25	3.25	**1.75**
	F_{mac}	3.42	2.50	5.58	4.17	3.75	**1.58**

Table 3 Performance comparison of the ALCS and the classifier-based AL methods using LinearSVM. (The relative rank of each algorithm is shown within the parentheses)

Datasets	Metrics	UBS	QBC	DWUS	MVAL	ALCS$_N$	ALCS$_D$
R15	*Acc*	57.78(4)	58.48(3)	52.16(6)	57.33(5)	58.98(2)	**59.07(1)**
	F_{mac}	**49.52(1)**	47.19(6)	47.68(3)	47.55(5)	47.56(4)	47.84(2)
Australian	*Acc*	83.10(2)	81.42(5)	82.52(4)	81.01(6)	82.97(3)	**86.99(1)**
	F_{mac}	82.92(3)	81.25(5)	82.41(4)	80.61(6)	82.94(2)	**86.92(1)**
Aggregation	*Acc*	77.87(3)	75.51(4)	64.42(6)	73.44(5)	80.15(2)	**85.53(1)**
	F_{mac}	51.01(4)	45.69(5)	36.59(6)	51.44(3)	52.54(2)	**53.36(1)**
Vehicle	*Acc*	61.91(6)	64.08(3)	62.96(5)	**66.12(1)**	63.32(4)	65.17(2)
	F_{mac}	60.55(6)	60.89(5)	61.54(3)	**63.47(1)**	61.25(4)	62.63(2)
Spambase	*Acc*	82.82(4)	85.81(2)	81.93(5)	79.36(6)	84.81(3)	**86.65(1)**
	F_{mac}	81.23(4)	84.69(2)	79.86(5)	79.26(6)	84.34(3)	**86.05(1)**
Waveforms	*Acc*	80.46(6)	82.52(3)	81.26(5)	82.32(4)	82.87(2)	**83.91(1)**
	F_{mac}	80.39(6)	82.53(3)	81.27(5)	82.35(4)	83.48(2)	**83.92(1)**
Electricity	*Acc*	97.25(3)	97.34(2)	95.63(5)	94.32(6)	97.17(4)	**98.87(1)**
	F_{mac}	96.99(2)	96.97(3)	95.24(5)	93.55(6)	96.92(4)	**98.77(1)**
DLA0.01	*Acc*	73.02(5)	84.56(4)	71.64(6)	85.13(2)	84.76(3)	**85.19(1)**
	F_{mac}	75.37(5)	84.33(2)	64.78(6)	**84.71(1)**	77.71(4)	78.31(3)
Penbased	*Acc*	83.36(5)	86.69(2)	83.03(6)	84.54(4)	86.19(3)	**88.78(1)**
	F_{mac}	83.02(5)	86.37(2)	82.45(6)	84.03 (4)	86.01(3)	**88.68(1)**
GasSensor	*Acc*	78.29(4)	81.75(2)	77.63(5)	77.48 (6)	79.32(3)	**82.26(1)**
	F_{mac}	74.91(4)	77.41(2)	72.45(6)	73.32 (5)	75.56(3)	**79.85(1)**
DCCC	*Acc*	78.84(4)	79.36(2)	78.74(5)	76.86 (6)	79.05(3)	**79.61(1)**
	F_{mac}	51.56(5)	55.74(2)	50.52(6)	52.16 (3)	52.05(4)	**56.83(1)**
MINST	*Acc*	80.97(5)	82.41(3)	80.57(6)	81.55 (4)	84.06(2)	**84.42(1)**
	F_{mac}	80.92(5)	82.13(3)	80.49(6)	81.23 (4)	83.85(2)	**84.25(1)**
Mean ranks	*Acc*	4.25	2.92	5.33	4.58	2.83	**1.08**
	F_{mac}	4.17	3.33	5.08	4.00	3.08	**1.33**

clustering-based methods, are used in the experiments. For classifier-based AL approaches, the UBS [2], QBC [6], density weighted uncertainty sampling (DWUS) [10], and maximum variance for active learning (MVAL) [60] approaches are used. The UBS, QBC and DWUS are well-known *IBQ* methods and they are extensively studied in AL community. The MVAL method is a recently developed AL method that queries both the informative and representative samples. On the other hand, the QUIRE [17], ALEC [44], active learning through multi-standard optimization (MSAL) [46], active learning through label error statistical (ALSE) [47], and three-way active learning through clustering selection (TACS) [29] approaches are chosen as the representatives of clustering-based AL approaches. Among all compared clustering-based AL methods, ALSE and TACS are the most recent approaches and have shown substantial superiority over other methods. The MSAL is chosen as the representative of clustering-based AL methods with diversity exploration. To show the efficacy of the diversity exploration strategy for the proposed method, we experiment two versions of ALCS framework, including ALCS$_N$ (no diversity exploration) and ALCS$_D$ (with diversity exploration).

For UBS, QBC, DWUS, and QUIRE, experiments are conducted in Python 3.7.4 platform using the *libact* package [62]. In QBC, logistic regression, support vector machine and perceptron classifiers are used as committee members. With the source code from the authors, ALEC and TACS are experimented in JAVA. The python codes of ALSE and MSAL methods are also provided by the authors. We use the MATLAB code of MVAL from the authors for experiment purpose. The ALCS$_N$ and ALCS$_D$ are implemented using Python 3.7.4 language and the code is available at.[1]

5.3 Experimental setting

Two well-known evaluation metrics, including *Acc*, and *F-measure*, are used to compare all AL methods. To account

[1] https://github.com/XuyangAbert/ALCS

Table 4 Summary of the Friedman rank test for $\mathcal{F}_F(M = 6, N = 12)$ for UBS, QBC, DWUS, and ALCS. (M is the number of compared methods and N is the number of datasets)

Base Classifiers	Metric	$\mathcal{F}_F$	Critical value ($\alpha = 0.05$)
KNN	Acc	30.429	2.9961
	F_{mac}	27.762	
LinearSVM	Acc	32.667	2.9961
	F_{mac}	22.714	

Table 6 Summary of the Friedman rank test for $\mathcal{F}_F(M = 7, N = 12)$ for ALEC, QUIRE, MSAL, ALSE, TACS, ALCS$_\mathrm{N}$, and ALCS$_\mathrm{D}$

Metric	$\mathcal{F}_F$	Critical value ($\alpha = 0.05$)
Acc	42.036	2.913
F_{mac}	40.893	

for imbalanced class distribution in experimental datasets, the macro-average of the *F-measures* is used as defined below.

$$F_{mac} = \frac{1}{n_c}\sum_{i=1}^{n_c} F_i, \tag{17}$$

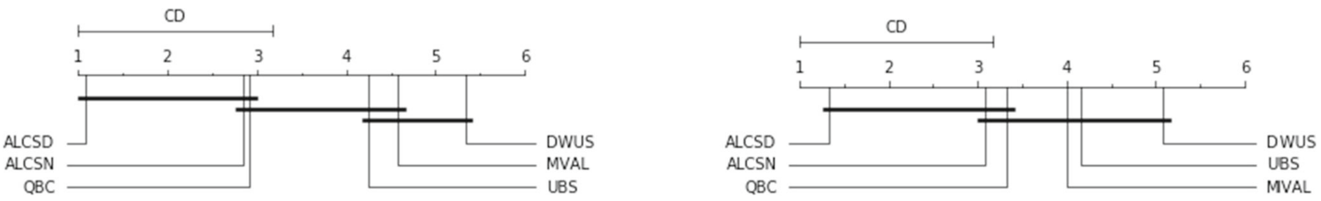
where F_i denotes the *F-measure* for the i^{th} class.

The k-nearest-neighbor (kNN) [1] and linear support vector machine (LinearSVM) [5] classifiers are used to evaluate the quality of the queried samples selected by the classifier-based approaches, ALCS$_\mathrm{N}$, and ALCS$_\mathrm{D}$. The value of k is set to 3 for the kNN classifier in all experiments. Following the experimental design in [10, 45], the number of queried labels n_q usually ranges from $0.05n_U$ to $0.1n_U$. We set n_q to be $0.1n_U$ in datasets with less than 1000 samples and n_q is set as $0.05n_U$ for the remaining datasets. An initial label set with a size of $\frac{n_q}{2}$ samples is randomly drawn from each class to train the classifiers for classifier-based AL approaches such that the initial training set includes samples from all classes for each dataset.

Considering the randomness from the selection of the initial label set for classifier-based AL approaches,

Table 5 Performance comparison of ALCS and five clustering-based AL methods

Dataset	Metrics	ALEC	QUIRE	MSAL	ALSE	TACS	ALCS$_\mathrm{N}$	ALCS$_\mathrm{D}$
R15	Acc	84.58(7)	**99.26(1)**	99.14(2)	86.27(6)	98.45(5)	98.93(4)	99.07(3)
	F_{mac}	84.09(7)	**99.21(1)**	98.27(4)	83.94(6)	97.66(5)	98.61(3)	99.06(2)
Australian	Acc	80.80(6)	81.29(5)	68.78(7)	81.38(4)	82.08(3)	82.14(2)	**83.31(1)**
	F_{mac}	79.71(6)	80.87(4)	68.69(7)	80.82(5)	80.92(3)	81.99(2)	**83.06(1)**
Aggregation	Acc	91.06(6)	71.01(7)	91.25(5)	91.91(3)	92.74(2)	91.82(4)	**99.43(1)**
	F_{mac}	76.86(6)	44.21(7)	76.92(5)	77.63(3)	78.15(2)	77.51(4)	**99.09(1)**
Vehicle	Acc	46.11(7)	53.23(4)	48.92(5)	46.39(6)	53.45(2)	53.34(3)	**54.74(1)**
	F_{mac}	54.66(3)	49.52(7)	**55.12(1)**	52.37(6)	54.83(2)	52.78(5)	54.30(4)
Spambase	Acc	76.48(5)	75.73(6)	75.32(7)	76.57(4)	**82.91(1)**	79.58(3)	81.54(2)
	F_{mac}	75.85(4)	74.79(7)	75.87(6)	75.46(5)	80.28(2)	79.08(3)	**80.92(1)**
Waveforms	Acc	75.42(6)	75.87(5)	75.32(7)	76.89(3)	78.17(1)	75.98(4)	76.66(2)
	F_{mac}	75.84(4)	74.91(7)	75.47(5)	75.12(6)	76.52(2)	75.91(3)	**76.62(1)**
Electricity	Acc	82.81(6)	82.48(7)	83.01(4)	83.22(3)	82.88(5)	83.43(2)	**85.34(1)**
	F_{mac}	80.47(4)	79.89(7)	80.44(6)	80.83(3)	80.56(5)	81.05(2)	**83.76(1)**
DLA0.01	Acc	86.27(6)	72.14(7)	92.48(5)	93.18(4)	**99.22(1)**	93.46(3)	93.61(2)
	F_{mac}	86.28(6)	72.51(7)	86.98(5)	87.15(4)	**97.98(1)**	88.13(3)	88.26(2)
Penbased	Acc	87.94(6)	82.74(7)	89.48(4)	88.13(5)	91.24(3)	91.44(2)	**94.80(1)**
	F_{mac}	86.98(6)	72.68(7)	88.04(5)	89.01(4)	91.03(3)	91.47(2)	**94.76(1)**
GasSensor	Acc	64.94(6)	64.40(7)	65.79(5)	66.44(4)	66.88(3)	67.03(2)	**72.81(1)**
	F_{mac}	61.95(6)	60.60(7)	62.84(5)	63.74(4)	64.25(3)	64.39(2)	**71.55(1)**
DCCC	Acc	**76.88(1)**	75.15(6)	74.85(7)	75.26(5)	75.45(4)	76.36(3)	76.43(2)
	F_{mac}	54.16(5)	44.21(7)	49.56(6)	57.35(2)	54.95(4)	55.41(3)	**60.84(1)**
MNIST	Acc	87.58(5)	84.52(7)	87.12(6)	88.45(3)	87.75(4)	90.92(2)	**91.83(1)**
	F_{mac}	87.15(3)	83.48(7)	86.54(5)	86.81(4)	84.07(6)	89.75(2)	**91.79(1)**
Avg. ranks	Acc	5.58	5.75	5.33	4.17	2.83	2.83	**1.50**
	F_{mac}	5.00	6.25	5.00	4.33	3.17	2.83	**1.42**



(a) Acc with kNN (b) F_{mac} with kNN

Fig. 4 Comparison of ALCS against other classifier-based AL methods with the Nemenyi test with $\alpha = 0.05$ in terms of kNN

each experiment is repeated ten times to report the mean value. For ALEC, ALSE, MSAL, QUIRE, TACS, ALCS_N and ALCS_D methods, no randomness is involved and experiments are carried out without repetition. The comparison results are summarized in Tables 2, 3, and 4. Among all clustering-based approaches, ALEC, TACS, MSAL, and ALSE adopt a classification framework similar to the kNN classifier. Consequently, we use kNN as the base classifier for QUIRE, ALCS_N, and ALCS_D methods. The comparison with clustering-based methods is presented in Tables 5 and 6. In all tables, the best results are highlighted in boldface.

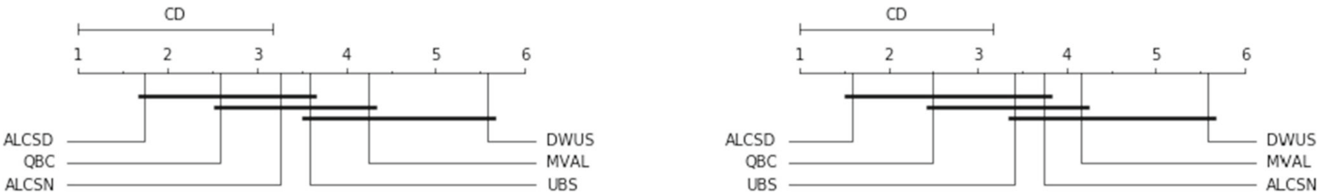
5.4 Comparison with classifier-based AL methods

As shown in Table 2, the proposed approach with diversity exploration procedure, namely ALCS_D, shows a better classification performance than the other four classifier-based AL methods for the majority of benchmark datasets using the kNN classifier. More concretely, in terms of Acc and F_{mac}, ALCS_D shows the highest average ranks of 1.75 and 1.58 on the kNN classifier, respectively. Without the diversity exploration procedure, ALCS_N has the third highest average rank on Acc and fourth highest average rank on F_{mac}. From these results, we can induce that the proposed diversity exploration strategy effectively leverages the performance of ALCS framework on kNN classifier.

With the LinearSVM classifier, Table 3 demonstrates that both the ALCS_D and ALCS_N methods have better performance than classifier-based AL methods. This is supported from the perspective that ALCS_D and ALCS_N are the top two ranked approaches in Table 3. Similar to the kNN classifier, Table 3 also shows that EDET effectively leverages the performance of the proposed AL method. Furthermore, it is observed that the proposed framework shows better classification performance than the existing classifier-based AL methods on the LinearSVM classifier. This can be attributed to the fact that LinearSVM tries to approximate a linearly separable boundary function between classes and the bi-cluster boundary-based selection helps to refine the boundary function by querying labels from the cross-boundary of two adjacent clusters.

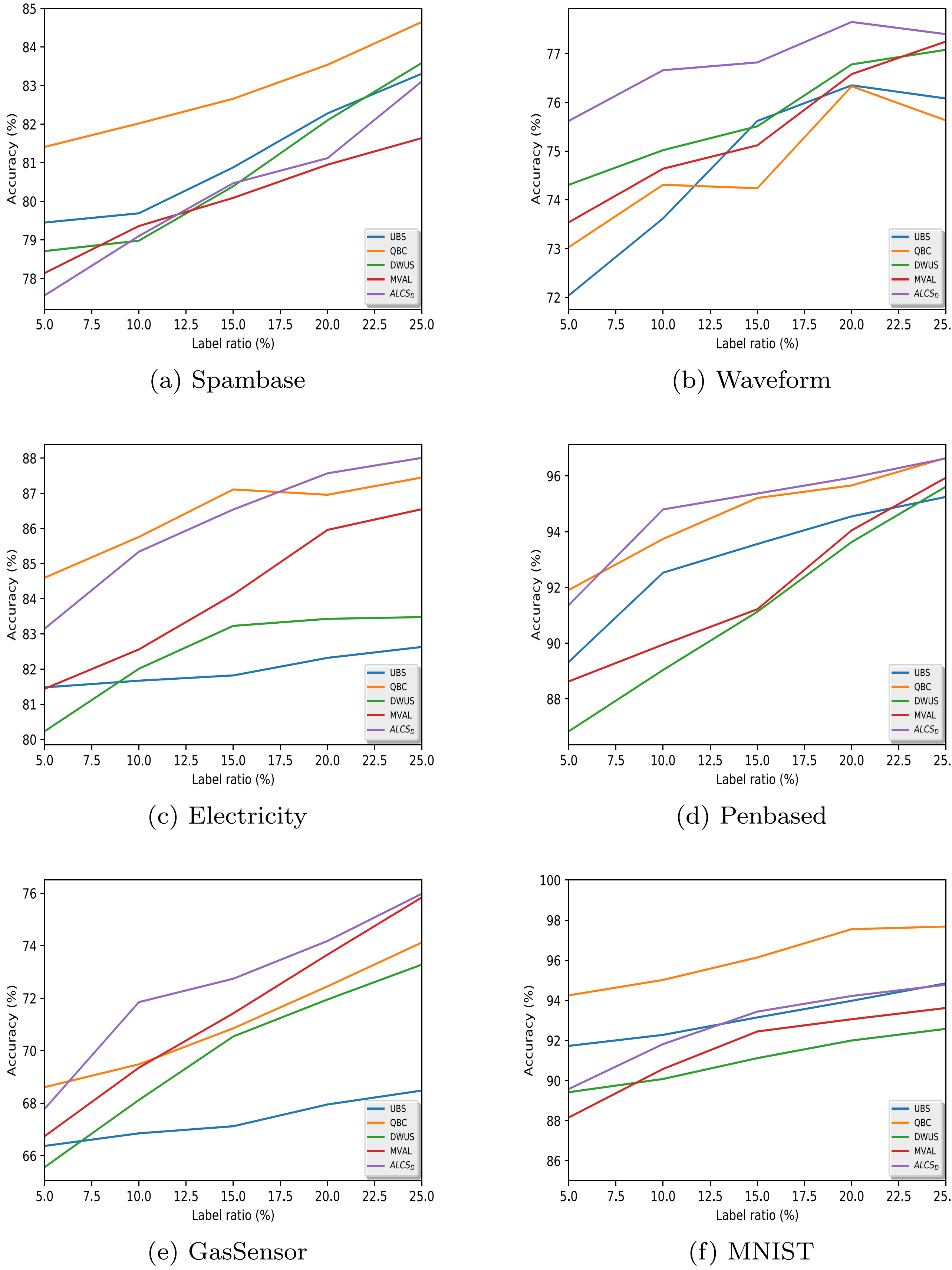
Table 4 shows the statistical analysis of the experimental results from the two versions of ALCS framework versus the other four classifier-based AL methods. Since ALCS_D has shown obvious superiority to the ALCS_N framework, we only discussed the statistical analysis between ALCS_D and other classifier-based AL methods. The non-parametric Friedman test [8] is used here to compare multiple methods based on their average ranks for the benchmark datasets. According to the Friedman test statistics, the null hypothesis that ALCS_D and ALCS_N have a similar performance with the other four classifier-based methods is rejected. Then, the Nemenyi post-hoc [8] test is performed with a significance level of $\alpha = 0.05$. The critical distance (CD) diagrams are presented in Figs. 4 and 5. For kNN classifier, Fig. 4 shows that ALCS_D has statistically better performance than all classifier-based AL methods except for QBC method. For QBC method, ALCS_D is statistically comparable.

In terms of Acc, Figs. 6 and 7 display the learning curves of ALCS_D and other four classifier-based AL methods among six benchmark datasets, which includes Spambase, Waveform, Electricity, Penbased, GasSensor and MNIST. These datasets have medium large sample size and high



(a) Acc with LinearSVM (b) F_{mac} with LinearSVM

Fig. 5 Comparison of ALCS against other classifier-based AL methods with the Nemenyi test with $\alpha = 0.05$ in terms of LinearSVM



(a) Spambase

(b) Waveform

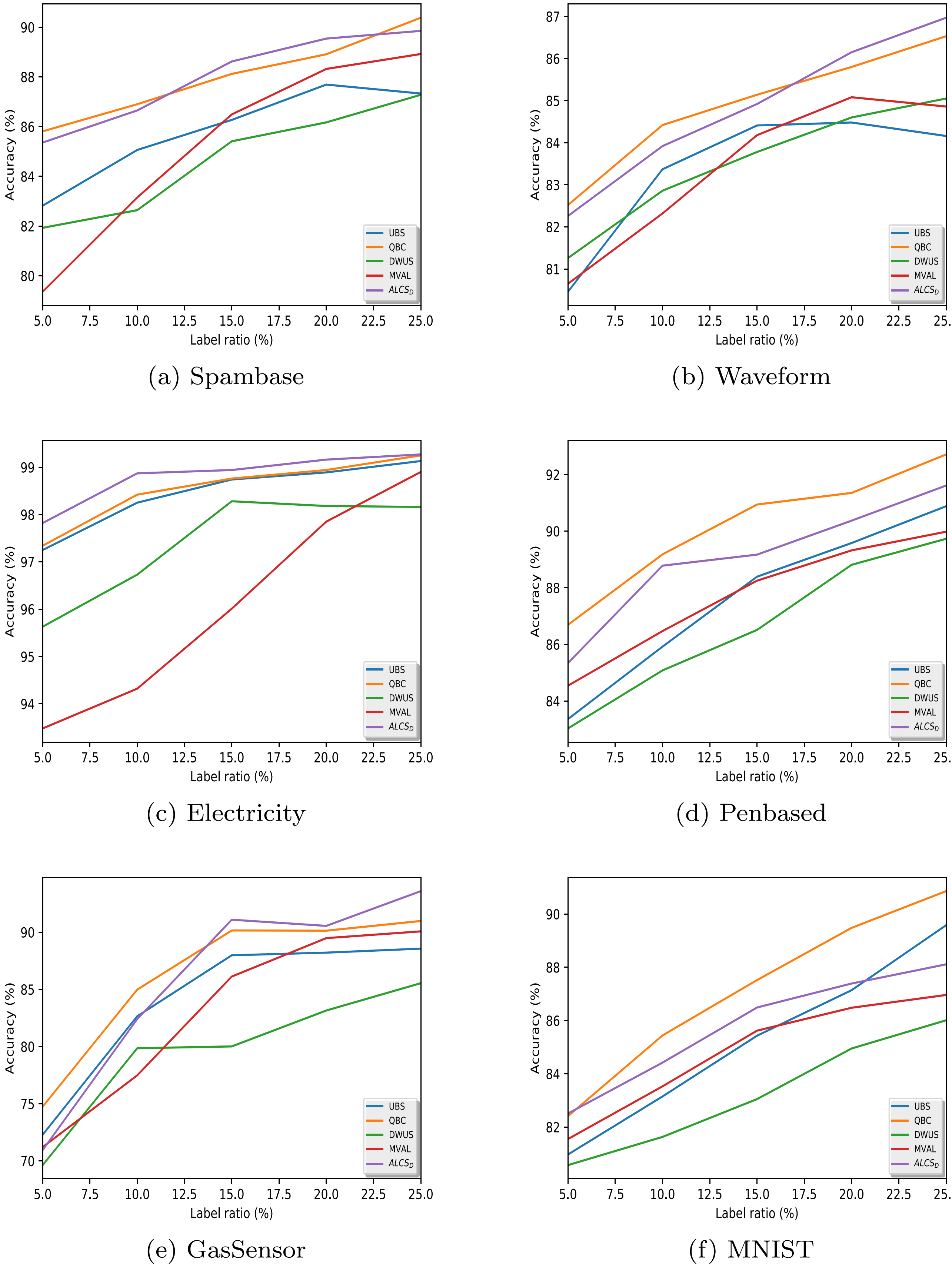
(c) Electricity

(d) Penbased

(e) GasSensor

(f) MNIST

Fig. 6 Learning cures of ALCS and four classifier-based AL methods as label ratio increases from 5% to 25% in terms of *k*NN



(a) Spambase

(b) Waveform

(c) Electricity

(d) Penbased

(e) GasSensor

(f) MNIST

Fig. 7 Learning cures of ALCS and four classifier-based AL methods as label ratio increases in terms of LinearSVM

dimensionality. The label ratio ranges from 5% to 25% with a step size of 5% and the learning curves are obtained using the average of ten runs for each method. From Figs. 6 and 7, ALCS$_{\mathrm{D}}$ always demonstrates better performance than the UBS, DWUS, and MVAL methods in both classifiers as the label ratio increases. Compared to the QBC method, ALCS$_{\mathrm{D}}$ achieves slightly better or comparable performance when the label ratio increases. This can be explained by the fact that QBC utilizes an ensemble of classifiers and the model performance improves significantly as more labeled data are available. Furthermore, the ALCS framework consistently shows good learning performance against the classifier-based AL methods on two different classification technique.

5.5 Comparison with clustering-based AL methods

Table 5 compares the performance of ALCS$_{\mathrm{N}}$ and ALCS$_{\mathrm{D}}$ with five clustering-based AL approaches. Due to the obvious superiority of ALCS$_{\mathrm{D}}$ over ALCS$_{\mathrm{N}}$ method, we focus on the comparison between ALCS$_{\mathrm{D}}$ and five state-of-the-art clustering-based methods. It is observed that ALCS$_{\mathrm{D}}$ provides a better performance in most datasets, and it has the highest average ranks for both Acc and F_{mac}. In Australian, Aggregation, Spambase, Waveforms, Electricity, Penbased, GasSensor, and MNIST datasets, ALCS$_{\mathrm{D}}$ outperforms the other five clustering-based AL methods on both Acc and F_{mac} metrics. These results imply the efficacy of ALCS$_{\mathrm{D}}$ in handling datasets with highly overlapped classes. Moreover, ALCS$_{\mathrm{D}}$ demonstrates better classification performance than the other five clustering-based AL methods on high-dimensional datasets such as Electricity, GasSensor, Spambase, and MNIST, which is shown in Table 5.

In Table 6, the Friedman test indicates that ALCS$_{\mathrm{D}}$ has statistically different classification performance than the other five clustering-based AL methods. Then, the Nemenyi post-hoc test is performed and the CD diagram is shown in Fig. 8. Figure 8 displays that ALCS$_{\mathrm{D}}$ is statistically better than ALEC, QUIRE, ALSE, and MSAL methods in terms of Acc and F_{mac}. On the other hand, ALCS$_{\mathrm{D}}$ presented statistically comparable performance with the TACS method, in which TACs method utilized an ensemble of clustering methods to explore the cluster structure. Overall, the comparison study shows that ALCS$_{\mathrm{D}}$ provides a statistically comparable or better performance than other clustering-based AL methods without tuning any clustering parameters.

5.6 Summary of discussions

Based on the experimental analysis presented, several important points of discussions are summarized regarding the four aforementioned questions:

(1) Regarding the efficacy of EDET, Tables 2 and 3 demonstrate that EDET effectively leverages the learning performance of the ALCS framework.

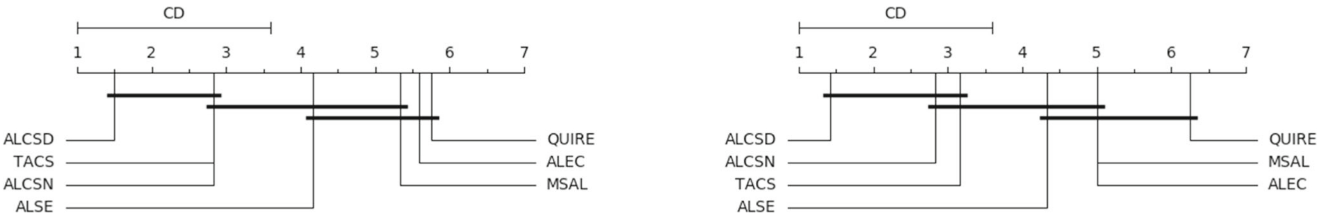
(2) According to Tables 2 and 3, ALCS can provide better or comparable performance than the classifier-based AL methods and it does not require any prior label information. From Figs. 4 and 5, the statistical analysis justifies that ALCS is statistically better, or comparable in relation to the classifier-based AL methods.

(3) From Figs. 6 and 7, ALCS always shows good learning performance against the classifier-based AL methods on two different types of classifiers such as kNN and LinearSVM. Similar observations are obtained from Tables 2 and 3.

(4) Compared with five existing clustering-based AL methods, Table 5 and Fig. 8 justify that ALCS statistically outperforms all methods except for the TACS method. For TAC method, ALCS achieves statistically comparable performance while TAC employs an ensemble of clustering methods.

6 Conclusion

In this paper, we proposed a novel active learning framework using clustering-based sampling to handle the lack of prior label information. It utilizes the FPS-clustering procedure to explore the structure of unlabeled



(a) Acc

(b) F_{mac}

Fig. 8 Comparison of ALCS against other clustering-based AL methods with the Nemenyi test with $\alpha = 0.05$

data without an exhaustive parameter tuning. To perform the *IBQ* and *RBQ* simultaneously, a new distance-based sample selection procedure was employed by combining the center-based and boundary-based selection methods. This hybrid sample selection procedure is capable of adjusting the sampling portion between *IBQ* and *RBQ* based on the density of each cluster. To improve the learning performance, we introduced a new bi-cluster boundary-based selection procedure to identify informative samples from the boundary region among adjacent clusters. A mathematical justification of the bi-cluster boundary-based selection procedure is provided. Furthermore, we developed an effective diversity exploration strategy to reduce the redundancy among queried samples. Experimental results established that ALCS provided statistically better or comparable performance than the four classifier-based AL methods, and it does not require an initial labeled dataset. The comparison results with clustering-based AL methods demonstrated that ALCS showed statistically better or comparable performance than the five clustering-based AL approaches without tuning the clustering parameters. ALCS shows better performance than other clustering-based AL methods in datasets with overlapped classes, while the statistical analysis does not indicate a significant difference. Therefore, the efficacy of ALCS in handling the overlapped classes needs further investigation.

Our study is impetus to the following future research.

- *Investigate the efficacy of ALCS in handling highly overlapped classes:* Despite that ALCS shows good performance in some datasets with highly overlapped classes, more extensive experiments will be conducted to further test the performance of addressing the overlap among classes.
- *Extension to online AL problems:* Instead of using the FPS-clustering method, the stream clustering method in [55] will be used to extend the ALCS framework to handle streaming data with no initial label information.

Acknowledgements This paper is based on research sponsored by the Air Force Research Laboratory and the Office of the Secretary of Defense (OSD) under agreement number FA8750-15-2-0116. Also, this work is partially funded through the National Science Foundation (NSF) under grant number 2000320.

Author Contributions The conceptualization, methodology, software implementation & debug, and validation are contributed by Xuyang Yan, Shabnam Nazmi, Biniam Gebru, and Mrinmoy Sarkar. The first draft was prepared by Xuyang Yan and all authors participated the editing of the manuscript. Xuyang Yan and Dr.Abodllah Homaifar shaped up the original idea of the new contribution for the first revision. The implementation of the new contribution and additional experiments are conducted by Xuyang Yan, Mrinmoy Sarkar and Kishor Datta Gupta. Drs. Mohd Anwar and Adbollah Homaifar suggested many important modifications to improve the overall writing quality and re-organzation of this revised manuscript. This research was supervised by Professors Abdollah Homaifar and Mohd Anwar. The funding of this research was acquired by Dr.Abdollah Homaifar.

Funding This study was funded by Air Force Research Laboratory and the Office of the Secretary of Defense (OSD) under agreement number FA8750-15-2-0116. This work is also partially supported by National Science Foundation under grant number 2000320.

Availability of data and material The data that support the findings of this study are available from the UCI machine learning repository [https://archive.ics.uci.edu/ml/index.php].

Code Availability The python code is developed by the authors and it is available at https://github.com/XuyangAbert/ALCS.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Altman NS (1992) An introduction to kernel and nearest-neighbor nonparametric regression. Amer Stat 46(3):175–185
2. Cai D, He X (2011) Manifold adaptive experimental design for text categorization. IEEE Trans Knowl Data Eng 24(4):707–719
3. Chattopadhyay R, Wang Z, Fan W, Davidson I, Panchanathan S, Ye J (2013) Batch mode active sampling based on marginal probability distribution matching. ACM Trans Knowl Discov Data (TKDD) 7(3):1–25
4. Cortes C, Mohri M (2014) Domain adaptation and sample bias correction theory and algorithm for regression. Theor Comput Sci 519:103–126
5. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
6. Dagan I, Engelson SP (1995) Committee-based sampling for training probabilistic classifiers. In: Machine Learning Proceedings 1995, Elsevier. pp 150–157
7. Dasgupta S, Hsu D (2008) Hierarchical sampling for active learning. In: Proceedings of the 25th international conference on Machine learning, pp 208–215
8. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7(Jan):1–30
9. Dheeru D, Karra Taniskidou E (2017) UCI machine learning repository. http://archive.ics.uci.edu/ml
10. Donmez P, Carbonell JG, Bennett PN (2007) Dual strategy active learning. In: European Conference on Machine Learning, Springer. pp 116–127
11. Freund Y, Seung HS, Shamir E, Tishby N (1997) Selective sampling using the query by committee algorithm. Mach Learn 28(2-3):133–168
12. Gu S, Cai Y, Shan J, Hou C (2019) Active learning with error-correcting output codes. Neurocomputing 364:182–191
13. Hoi SC, Jin R, Zhu J, Lyu MR (2006) Batch mode active learning and its application to medical image classification. In: Proceedings of the 23rd international conference on Machine learning, pp 417–424
14. Hoi SC, Jin R, Zhu J, Lyu MR (2009) Semisupervised svm batch mode active learning with applications to image retrieval. ACM Trans Inform Syst (TOIS) 27(3):1–29
15. Holub A, Perona P, Burl MC (2008) Entropy-based active learning for object recognition. In: 2008 IEEE Computer

Society Conference on Computer Vision and Pattern Recognition Workshops, IEEE. pp 1–8

16. Huang GB, Zhu QY, Siew CK (2006) Extreme learning machine: theory and applications. Neurocomputing 70(1-3):489–501
17. Huang SJ, Jin R, Zhou ZH (2010) Active learning by querying informative and representative examples. In: Advances in neural information processing systems, pp 892–900
18. Huang SJ, Zong CC, Ning KP, Ye HB (2021) Asynchronous active learning with distributed label querying. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, pp 2570–2576
19. Kading C, Freytag A, Rodner E, Bodesheim P, Denzler J (2015) Active learning and discovery of object categories in the presence of unnameable instances. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4343–4352
20. Kee S, Del Castillo E, Runger G (2018) Query-by-committee improvement with diversity and density in batch active learning. Inf Sci 454:401–418
21. Krempl G, Kottke D, Lemaire V (2015) Optimised probabilistic active learning (opal). Mach Learn 100(2):449–476
22. Lewis DD, Catlett J (1994) Heterogeneous uncertainty sampling for supervised learning. In: Machine learning proceedings 1994, Elsevier, pp 148–156
23. Lewis DD, Gale WA (1994) A sequential algorithm for training text classifiers. In: SIGIR'94, Springer. pp 3–12
24. Li H, Wang Y, Li Y, Xiao G, Hu P, Zhao R (2021a) Batch mode active learning via adaptive criteria weights. Appl Intell 51(6):3475–3489
25. Li H, Wang Y, Li Y, Xiao G, Hu P, Zhao R, Li B (2021b) Learning adaptive criteria weights for active semi-supervised learning. Inf Sci 561:286–303
26. Lu J, Zhao P, Hoi SC (2016) Online passive-aggressive active learning. Mach Learn 103(2):141–183
27. Lughofer E (2012) Hybrid active learning for reducing the annotation effort of operators in classification systems. Pattern Recogn 45(2):884–896
28. Lughofer E (2017) On-line active learning: a new paradigm to improve practical useability of data stream modeling methods. Inf Sci 415:356–376
29. Min F, Zhang SM, Ciucci D, Wang M (2020) Three-way active learning through clustering selection. Int J Mach Learn Cybern 11(5):1033–1046
30. Nguyen HT, Smeulders A (2004) Active learning using pre-clustering. In: Proceedings of the twenty-first international conference on Machine learning, p 79
31. Nuhu AR, Yan X, Opoku D, Homaifar A (2021) A niching framework based on fitness proportionate sharing for multi-objective genetic algorithm (moga-fps). In: Proceedings of the Genetic and Evolutionary Computation Conference Companion, Association for Computing Machinery, New York, NY, USA, GECCO '21, p 191–192. https://doi.org/10.1145/3449726.3459566
32. Rodriguez A, Laio A (2014) Clustering by fast search and find of density peaks. Science 344(6191):1492–1496
33. Roy N, McCallum A (2001) Toward optimal active learning through monte carlo estimation of error reduction. ICML, Williamstown 441–448
34. Schein AI, Ungar LH (2007) Active learning for logistic regression: an evaluation. Mach Learn 68(3):235–265
35. Settles B, Craven M, Ray S (2008) Multiple-instance active learning. In: Advances in neural information processing systems, pp 1289–1296
36. Seung HS, Opper M, Sompolinsky H (1992) Query by committee. In: Proceedings of the fifth annual workshop on Computational learning theory, pp 287–294
37. Smith JS, Nebgen B, Lubbers N, Isayev O, Roitberg AE (2018) Less is more: Sampling chemical space with active learning. J Chem Phys 148(24):241733
38. Tang YP, Huang SJ (2021) Dual active learning for both model and data selection. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, pp 3052–3058
39. Tong S, Koller D (2001) Support vector machine active learning with applications to text classification. J Mach Learn Res 2(Nov):45–66
40. Tsou YL, Lin HT (2019) Annotation cost-sensitive active learning by tree sampling. Mach Learn 108(5):785–807
41. Viering TJ, Krijthe JH, Loog M (2019) Nuclear discrepancy for single-shot batch active learning. Mach Learn 108(8):1561–1599
42. Wang L, Hu X, Yuan B, Lu J (2015) Active learning via query synthesis and nearest neighbour search. Neurocomputing 147:426–434
43. Wang M, Hua XS (2011) Active learning in multimedia annotation and retrieval: a survey. ACM Trans Intell Syst Technol (TIST) 2(2):1–21
44. Wang M, Min F, Zhang ZH, Wu YX (2017a) Active learning through density clustering. Expert Syst Appl 85:305–317
45. Wang M, Fu K, Min F (2018a) Active learning through two-stage clustering. In: 2018 IEEE International conference on fuzzy systems (FUZZ-IEEE), IEEE, pp 1-7
46. Wang M, Zhang YY, Min F (2019) Active learning through multi-standard optimization. IEEE Access 7:56772–56784
47. Wang M, Fu K, Min F, Jia X (2020) Active learning through label error statistical methods. Knowl-Based Syst 189:105140
48. Wang R, Wang XZ, Kwong S, Xu C (2017b) Incorporating diversity and informativeness in multiple-instance active learning. IEEE Trans Fuzzy Syst 25(6):1460–1475
49. Wang Z, Ye J (2015) Querying discriminative and representative samples for batch mode active learning. ACM Trans Knowl Discov Data (TKDD) 9(3):1–23
50. Wang Z, Du B, Zhang L, Zhang L (2016) A batch-mode active learning framework by querying discriminative and representative samples for hyperspectral image classification. Neurocomputing 179:88–100
51. Wang Z, Fang X, Tang X, Wu C (2018b) Multi-class active learning by integrating uncertainty and diversity. IEEE Access 6:22794–22803
52. Workineh A, Homaifar A (2012) Fitness proportionate niching: Maintaining diversity in a rugged fitness landscape. In: Proceedings of the International Conference on Genetic and Evolutionary Methods (GEM), The Steering Committee of The World Congress in Computer Science Computer ..., pp 1-7
53. Xiao Y, Chang Z, Liu B (2020) An efficient active learning method for multi-task learning. Knowl-Based Syst 190:105137
54. Yan X, Homaifar A, Nazmi S, Razeghi-Jahromi M (2017) A novel clustering algorithm based on fitness proportionate sharing. In: Systems, man, and cybernetics (SMC), 2017 IEEE International Conference on IEEE, pp 1960–1965
55. Yan X, Razeghi-Jahromi M, Homaifar A, Erol BA, Girma A, Tunstel E (2019) A novel streaming data clustering algorithm based on fitness proportionate sharing. IEEE Access 7:184985–185000
56. Yan X, Nazmi S, Erol BA, Homaifar A, Gebru B, Tunstel E (2020) An efficient unsupervised feature selection procedure through feature clustering. Pattern Recognition Letters
57. Yan X, Homaifar A, Sarkar M, Girma A, Tunstel E (2021) A clustering-based framework for classifying data streams. In: Proceedings of the Thirtieth International Joint Conference on

Artificial Intelligence IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, pp 3257–3263

58. Yang MS, Wu KL (2004) A similarity-based robust clustering method. IEEE Trans Pattern Anal Mach Intell 26(4):434–448
59. Yang Y, Loog M (2016) Active learning using uncertainty information. In: 2016 23Rd international conference on pattern recognition (ICPR), IEEE, pp 2646–2651
60. Yang Y, Loog M (2018) A variance maximization criterion for active learning. Pattern Recogn 78:358–370
61. Yang Y, Ma Z, Nie F, Chang X, Hauptmann AG (2015) Multi-class active learning by uncertainty sampling with diversity maximization. Int J Comput Vis 113(2):113–127
62. Yang YY, Lee SC, Chung YA, Wu TE, Chen SA, Lin HT (2017) libact: Pool-based active learning in python. arXiv:171000379
63. Yu D, Varadarajan B, Deng L, Acero A (2010) Active learning and semi-supervised learning for speech recognition: a unified framework using the global entropy reduction maximization criterion. Comput Speech Lang 24(3):433–444
64. Yu H, Sun C, Yang W, Yang X, Zuo X (2015) Al-elm: One uncertainty-based active learning algorithm using extreme learning machine. Neurocomputing 166:140–150

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Xuyang Yan[1] · Shabnam Nazmi[1] · Biniam Gebru[1] · Mohd Anwar[1] · Abdollah Homaifar[1] · Mrinmoy Sarkar[1] · Kishor Datta Gupta[1]

Xuyang Yan
xyan@aggies.ncat.edu

Shabnam Nazmi
snazmi@aggies.ncat.edu

Biniam Gebru
btgebru@aggies.ncat.edu

Mohd Anwar
manwar@ncat.edu

Mrinmoy Sarkar
msarkar@aggies.ncat.edu

Kishor Datta Gupta
gkishordatta@ncat.edu

[1] North Carolina A&T State University,
Greensboro, 27401, NC, USA