IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 16, 2020, accepted January 17, 2021, date of publication January 20, 2021, date of current version April 8, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3053003

Batch Mode Active Learning Based on Multi-Set Clustering

YAZHOU YANG[1,*], XIAOQING YIN[1,*], YANG ZHAO[1], JUN LEI[2], WEILI LI[2], AND ZHE SHU[1]
[1]College of Advanced Interdisciplinary Studies, National University of Defense Technology, Changsha 410073, China
[2]College of Systems Engineering, National University of Defense Technology, Changsha 410073, China

Corresponding author: Zhe Shu (shuzhe@nudt.edu.cn)

*Yazhou Yang and Xiaoqing Yin contributed equally to this work.

This work was supported by the National Natural Science Foundation of China under Grant 61906208.

ABSTRACT Batch mode active learning, where a batch of samples is simultaneously selected and labeled, is a challenging task. The challenge lies in how to maintain the informativeness and keep the diversity of selected samples concurrently. We propose a novel batch mode active learning that balances the informativeness and representativeness using multi-set clustering. Our method utilizes a sequential active learner to retain the informativeness by providing a ranking of unlabeled samples and constructing multiple informative sets for the subsequent clusterings. K-means clustering is used to minimize the redundancy among these samples and to improve the representativeness. Finally, the optimal batch chosen is the one minimizing the expected predictive variance on all the data. Our experimental results on a large number of benchmark datasets demonstrate excellent performance of the proposed method in comparison with current state-of-the-art batch mode active learning approaches.

INDEX TERMS Active learning, batch mode active learning.

I. INTRODUCTION

Recently, we have witnessed a sharp increase of the amount of training data used in classification or regression tasks. Though the availability of large input data tends to boost the performance of machine learning models, it also leads to a big challenge: manually labeling these hundreds of thousands of samples is very time-consuming and expensive [30]. This situation is even more serious in the field where manually categorizing these instances requires some experts in its own field, such as in the field of medical image classification [38]. Active learning has been proposed to tackle this challenge by querying the most informative subsets from the whole data and maintaining good learning performance.

Extensive studies have been undertaken for myopic active learning (MAL) where a single instance is queried at a time. However, few efforts have been devoted to batch mode active learning (BMAL) where a batch of samples is selected and labeled simultaneously. The advantage of BMAL over MAL is that BMAL does not need to retrain the model many times during a single selection step and is more suitable on some parallel labeling platforms, such as Amazon Mechanical Turk. However, there also exists several challenges for BMAL. The first one is that selecting k samples from a pool of n instances may lead to computational complications as the number of possible batches C_k^n can be very large, depending on the values of n and k of course. The second challenge lies in the formulation of an appropriate criterion to measure the overall information carried by of a batch of samples. Simply using a myopic selection criterion often leads to poor performance since it disregards the redundancy among selected instances.

In this paper, we focus on pool-based active learning where little labeled data and a large pool of unlabeled data are available. We propose a new pool-based batch mode active learning algorithm, namely Active Learning using Multi-set Clustering (ALMC). Different from typical clustering-based approaches which use a fixed number of samples for clustering, ALMC first applies a clustering model on multiple informative sets and then adopts a selection criterion to choose the optimal batch. More specifically, our method employs a myopic active learning algorithm to rank these unlabeled data and conducts K-means clustering to the top ranked subsets of multiple sizes. For each set, we select one best performing instance from each cluster to form a batch of k queried samples. After obtaining a number of batch solutions from multiple sets, we select the optimal one by measuring the overall predictive variance using the transductive experimental design (TED) criterion [39]. In brief, ALMC measures the informativeness of selected samples by using a myopic active

The associate editor coordinating the review of this manuscript and approving it for publication was Xin-Lin Huang.

learner, reduces the redundancy by K-means clustering and maintains the representativeness with optimal batch selection.

In this work, we make the following contributions:

- We propose a multi-set clustering based batch mode active learning which selects informative and representative samples with minimum redundancy.
- The proposed batch method can, in principle, be used in combination with any myopic active learner, giving good performance if the myopic learner is good.
- We carry out experiments on 52 benchmark datasets (both binary and multi-class datasets). The results clearly show the improvements our methods gives over the state of the art.

II. RELATED WORK

Active learning has received much attention in recent years. Many myopic active learning approaches have been well studied, such as query-by-committee [31], uncertainty sampling [20], [32], error reduction [28], model change [12], variance reduction [29] and variance maximization [36]. There exists a straightforward approach which extends myopic active learning to the batch setting by querying the top k samples based on its own criterion. However, this approach typically performs poor since it ignores the information overlap among the selected instances [30].

Existing batch mode active learning algorithms can be roughly divided into four categories: clustering-based methods, optimal experimental design, exploration-exploitation approaches, and the remaining algorithms which formulate batch selection as some combinatorial optimization problems. Clustering-based methods typically select top m samples ($m > k$), feed these candidates to a clustering algorithm (e.g., K-means) and eventually choose one instance from each cluster [9], [26]. A shortcoming of these approaches is that their performances are sensitive to a fixed parameter m, which value is hard to set.

Optimal experimental design, which is not specially designed for BMAL, can be used for BMAL since it can select multiple samples simultaneously without knowing their labels. For example, transductive experimental design (TED) selects the examples which leads to a minimum prediction variance on the validation data [39]. However, since it does not utilize label information, it mainly preserves representativeness and hardly captures informativeness.

The third class consists of a large collection of batch mode active learning algorithms which have the same characteristic: the aim is to select samples which achieve a balance between informativeness and representativeness by using a trade-off parameter, such as [2], [3], [5], [10], [16], [22], [35], [37], [33], [34]. For example, [37] proposed a multi-class BMAL by combining uncertainty sampling and diversity with a trade-off parameter. Similarly, [3] used KL-divergence to measure the redundancy among a batch of examples and combine the redundancy and the uncertainty. The key differences among these BMAL approaches are the criteria used to evaluate the informativeness and representativeness. They also share a common weakness: their performances are very sensitive to the choice of the trade-off parameter. How to set this parameter is a challenge referred to as the exploration-exploitation dilemma.

The fourth class is composed of various approaches which deal with batch selection using some sophisticated optimization techniques, such as [6], [13]–[15], [23], [1], [27], [40]. For example, [15] adopted the Fisher information matrix as a measure of the overall uncertainty and formulated it as a submodular optimization problem. Reference [13] put forward an NP-hard combinatorial optimization problem to query a batch of examples. Reference [6] viewed the batch selection as an adaptive submodular problem and proposed a greedy solution. However, the drawback of this kind of algorithms is that it usually requires some relaxations and may converge to local optima.

III. THE PROPOSED METHOD

In this section, a detailed description of the proposed method is presented. We start with the basic setting of pool-based batch mode active learning. We have labeled data $\mathbf{L} = \{x_i, y_i\}_{i=1}^{n_l}$, where $x_i \in \mathbb{R}^d$ is a feature vector and y_i is the label of x_i. In addition, a large pool of unlabeled examples $\mathbf{U}$ of size n_u is also available. In each iteration, k samples will be selected from the unlabeled pool $\mathbf{U}$ and labeled by human experts. $\mathbf{L}$ and $\mathbf{U}$ will be correspondingly updated.

Figure 1 gives an overview of the proposed method. The core of ALMC lies in the multi-set clustering. This is motivated by the fact that typical clustering-based approaches use a fixed number of examples (e.g., $m = 3k$ in [25]) for clustering and selection. This is non-adaptive during the whole active learning process. Our main idea is to apply the clustering algorithm on multiple informative sets of varied sizes m (e.g., $m \in \{k, 2k, 3k, \ldots\}$). Each set will offer a feasible solution of the selection of k samples. We choose the optimal batch among these solutions and present the selected samples to human annotators.

ALMC goes through the following steps: 1.

1) Train myopic active learner. ALMC trains a myopic active learner on the labeled data $\mathbf{L}$. This learner can be any of myopic active learning algorithms which can produce a ranking of the unlabeled samples. The algorithm helps determine the informativeness of the individual samples.
2) Ranking and clustering. The examples in $\mathbf{U}$ are ranked from most informative to least informative according to the myopic learner. The top m samples are chosen and fed to the clustering algorithm (e.g., K-means). ALMC repeats this procedure for different $m = \{k, 2k, 3k, \ldots, tk\}$ where t is a pre-defined value depending on the size n_u. Within each cluster, the sample with highest ranking will be chosen as the candidate of one batch. The clustering algorithm is used to reduce the redundancy among selected samples.
3) Batch selection. The multi-set clustering provides multiple solutions of a batch of selected samples.

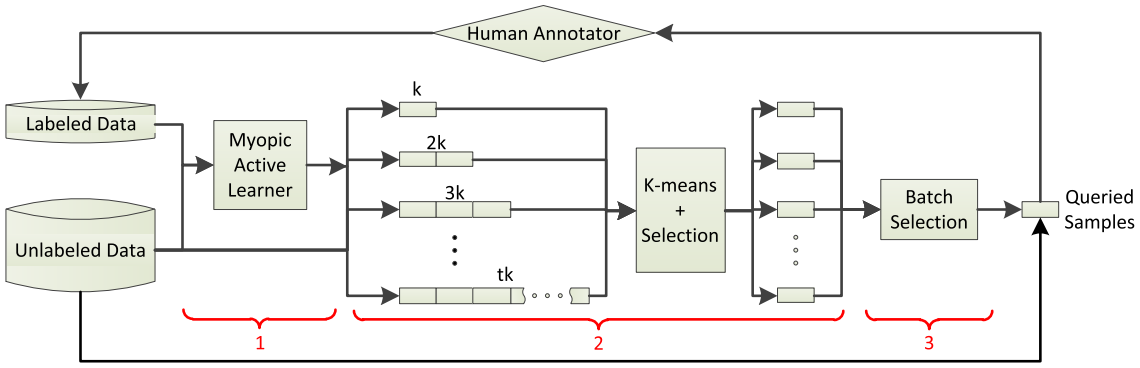


FIGURE 1. Overview of the proposed method ALMC. (1) First, ALMC trains one myopic active learner which is able to sort unlabeled examples by its own criterion in descending (or ascending) order. (2) Then, according to the ranking order, ALMC chooses multiple informative sets of size $\{k, 2k, 3k, \ldots, tk\}$ where the K-means clustering is conducted. (3) ALMC chooses an optimal batch by minimizing the expected predictive variance. Finally, human annotator will category these selected samples and the pool is updated.

ALMC chooses one of these batches as the final submitted batch by evaluating the expected predictive variance on both labeled and unlabeled data. The batch selection maintains the representativeness of queried instances.

In the following subsections, we will go through the specifics of the different steps.

A. MYOPIC ACTIVE LEARNER

The myopic active learner used in our method is very important since it also selects the most informative sample within each cluster. The quality of this learner has a clear impact on the performance of our method. This will be verified in the experimental section.

Recently, Yang and Loog [36] proposed a new myopic active learning method, namely Maximizing Variance for Active Learning (MVAL). Because of its overall good performance, we decided to take MVAL as our primary myopic active learner. We briefly present the main idea of MVAL.

MVAL has some similarity to classical query-by-committee (QBC) [31]. It forms a committee that consists of models trained on currently labeled data $\mathbf{L}$ and each unlabeled sample from $x_j \in \mathbf{U}$ with all possible labels y_j, i.e., $\mathbf{L} \cup \{x_j, y_j\}$. MVAL trains each committee member and records the posterior probabilities of unlabeled samples to form so-called retraining information matrices (RIMs). These RIMs are used to compute the disagreement among each committee member on all unlabeled samples. MVAL estimates two kinds of variance to evaluate the informativeness and representativeness and fuses these variances as a measure of the disagreement. Finally, MVAL will query the sample with maximum disagreement.

B. RANKING AND CLUSTERING

Given one certain myopic active learner, all the unlabeled instances are sorted from the most informative to least informative. Each time we pick the top $m = i \times k, i = 1, 2, \ldots t$ samples and apply the K-means clustering algorithm on these samples. Please note that we empirically set $t = \lceil \frac{n_u}{2k} \rceil$ in our method, where n_u is the number of unlabeled instances in $\mathbf{U}$ and $\lceil \cdot \rceil$ denotes the ceiling function. This means that ALMC only considers the samples which rank in the top half. This is reasonable because (1) the samples ranked in the bottom half are non-informative and (2) clustering on data sets of larger size m is unnecessarily time-consuming.

After obtaining the clustering result of K-means, we need to choose one candidate from each cluster to form a batch of queried samples of size k. There are some possible approaches, such as selecting the instances nearest to the centroid or randomly choosing one example. In our method, ALMC chooses the most informative example within each cluster. This is a balance of the representativeness and informativeness: K-means makes the selected samples diverse while selecting the top-ranked example preserves the informativeness as much as possible.

Note that the number of clusters to be detected by K-means is set to the batch size k. This means that in the case of $m = k$, i.e. the top k samples being fed into the K-means clustering method, each sample is treated as a cluster and eventually selected to form a batch query. In other words, the top k samples are always picked up to contract a batch candidate and then are fed into the batch selection stage.

C. BATCH SELECTION

Our multi-set clustering produces a limited number of promising batches. The next step is to choose one batch that we are going to present to the human annotator. Here we propose to use Transductive Experimental Design (TED) [39] to choose the best batch. TED is designed to select samples to minimize the average predictive variance on both labeled and unlabeled data. In other words, TED chooses examples which can well represent the remaining data with minimal reconstruction error. In brief, TED finds an approximate solution to the following optimization problem:

$$\min_{\mathbf{X} \subset \mathbf{V}} \quad \mathrm{Tr}(\mathbf{V}(\mathbf{X}^T\mathbf{X} + \mu\mathbf{I})^{-1}\mathbf{V}^T) \tag{1}$$

where $\mathrm{Tr}(\cdot)$ is the trace, $\mathbf{X} \in \mathbb{R}^{(n_l+k)\times d}$ denotes the samples to be queried, and $\mathbf{V} \in \mathbb{R}^{(n_l+n_u)\times d}$ represents all the data in pool,

including labeled and unlabeled data. μ is the regularization parameter and $\mathbf{I}$ is the identity matrix.

Though the K-means algorithm can make the selected examples distinct from each other, it can not guarantee that these lead to a small predictive variance on validation data. Therefore, ALMC utilizes the TED criterion to select an optimal batch which leads to a minimal predictive variance on all the data. Selecting such a batch of samples can promote the representativeness.

We denote the multi-set clustering results $\mathbf{C}_s$, $s = \{1, 2, \ldots, t\}$. Each $\mathbf{C}_s$ consists of k samples from $\mathbf{U}$. We denote $\mathbf{X}_s = [\mathbf{L}; \mathbf{C}_s]$.

$$\begin{aligned} &\arg\min_{s} \quad \mathrm{Tr}(\mathbf{V}(\mathbf{X}_s^T\mathbf{X}_s + \mu\mathbf{I})^{-1}\mathbf{V}^T) \\ &s.t. \quad\quad s \in \{1, 2, \ldots, t\} \end{aligned} \tag{2}$$

According to Equation 2, ALMC will select the optimal batch s^* and choose $\mathbf{C}_{s^*}$ as the final queried samples.

IV. EXPERIMENTS

In this section, we test the empirical performance of the proposed method and compare it against state-of-the-art batch mode active learning approaches.

A. DATASETS AND EXPERIMENTAL SETUP

Experiments are conducted on 52 benchmark datasets, including 40 two-class datasets and 12 multi-class datasets. Most of them come from the UCI Machine Learning Repository [21]. Three datasets, 3vs5, 5vs8 and 7vs9, are taken from the MNIST handwritten digit dataset [19]. The USPS dataset [17] is another handwritten digit database used in our experiment. We use GIST features [24] for scene13 dataset [11] and HOG features [8] for CIFAR10 dataset [18]. For computational efficiency, sub-sampling and principal component analysis (PCA) are applied on some large datasets to reduce the sample size and feature dimensionality. The detailed information of these preprocessed test datasets after pre-processing is listed in Table 1.

In our experiments, each dataset is randomly divided into training and test sets of equal size. For binary datasets, we set the initial labeled set size at k. For multi-class tasks, we randomly select one sample from each class. We repeat the experiments 10 times on each dataset and report the average performance. The batch size k is fixed at 5 in all the experiments.

We compare our method ALMC with random sampling (RS) and the following batch mode active learning algorithms:

- US: Uncertainty sampling with maximum entropy, which selects the top k examples with highest entropy.
- TED: Transductive Experimental Design [39].
- Fisher: It selects a batch of samples to reduce the Fisher information as much as possible [15].
- MinMax: It trains a semi-supervised SVM on both labeled and unlabeled data and balances the uncertainty and diversity via a min-max framework [16].
- USDM: Uncertainty sampling with diversity maximization, which retains the uncertainty and maximizes the diversity simultaneously [37].
- BatchRank: It balances the informativeness and diversity and offers some relaxations to solve the optimization problem [3].
- ALSC: A variant of our method ALMS, which reduces the multi-set clustering to a single set $m = 3k$ (similar to [26]). This will give us an idea of what the impact of our batch selection strategy is.

For the parameters used in the compared algorithms, we use the suggested value in the original paper. If no recommended value is provided (e.g., the trade-off parameter in USDM), we empirically tune these parameters to obtain good average performance over all the test datasets. For ALMC, the regularization parameter μ for TED is empirically set at 1. For fairness, linear SVM with the same parameter setting is used to evaluate the performance of all active learning algorithms. We use the LIBSVM package [4] and empirically set the regularization parameter $C = 10$. We use the area under the learning curve (ALC) [7] as the evaluation measure. More specifically, we first plot the learning curve, which shows the classification accuracy as a function of the number of queried instances (see Figures 3 and 4, for example), and then compute the area under this curve.

B. RESULTS

Tables 2 and 3 present the performance of batch mode active learning algorithms on binary and multi-class datasets, respectively. Since Fisher and MinMax are specifically designed for binary tasks, these two methods are omitted in the comparison on multi-class datasets. The average ALC and the ranking over all the datasets are also reported. On each dataset, we highlight the algorithms which perform the best or obtain comparable performance based on the paired t-test at a 95% significance level. They are emphasized in bold face and surrounded by a framed box. ''Win Counts'' shows the times an algorithm reaches the best or comparable performance. Figures 3 and 4 show the plots of average accuracy *w.r.t* the number of batch selection on binary and multi-class datasets, respectively.

We can see that the proposed method ALMC achieves overall best performance in comparison with the competing methods. ALMC performs the best in terms of the average performance measures on both binary and multi-class tasks. It obtains the highest average ALC score 0.832 and achieves the lowest average rank 2 on binary datasets. Meanwhile, it also succeeds in terms of Win Counts, performing well on 31 datasets from the total 40 binary test sets. ALSC, a single set ($m = 3$) version of ALMC, also achieves favourable performances on most datasets, but slightly worse. The latter still significantly outperforms former according to a paired t-test at the 95% significance level. Figure 5 shows the difference

TABLE 1. Characteristics of the preprocessed test datasets: the number of instances (# Ins), the feature dimensionality (# Fea) and the number of class (#C).

Dataset	(#Ins, #Fea, #C)	Dataset	(#Ins, #Fea, #C)	Dataset	(#Ins, #Fea, #C)
ac-inflam	(120, 6, 2)	acute	(120, 6, 2)	australian	(690, 14, 2)
blood	(748, 4, 2)	breast	(683, 10, 2)	credit	(690, 15, 2)
cylinder	(512, 35, 2)	diabetes	(768, 8, 2)	fertility	(100, 9, 2)
german	(1000, 24, 2)	heart	(270, 13, 2)	hepatitis	(255, 19, 2)
hill	(606, 100, 2)	ionosphere	(351, 34, 2)	liver	(345, 6, 2)
mushrooms	(1000, 112, 2)	mammographic	(961, 5, 2)	musk1	(476, 166, 2)
ooctris2f	(912, 25, 2)	ozone	(1000, 72, 2)	parkinsons	(195, 22, 2)
pima	(768, 8, 2)	planning	(182, 12, 2)	sonar	(208, 60, 2)
splice	(1000, 60, 2)	tictactoe	(958, 9, 2)	vc2	(310, 6, 2)
vehicle	(435, 18, 2)	wine	(178, 13, 2)	wisc	(699, 9, 2)
wdbc	(569, 31, 2)	DvsP	(1608, 16, 2)	EvsF	(1543, 16, 2)
IvsJ	(1502, 16, 2)	MvsN	(1575, 16, 2)	VvsY	(1577, 16, 2)
UvsV	(1550, 16, 2)	3vs5	(1500, 784, 2)	5vs8	(1500, 784, 2)
7vs9	(1500, 784, 2)	car	(900, 6, 4)	contrac	(1473, 9, 3)
heart_cleveland	(303, 13, 5)	led_display	(1000, 7, 10)	pendigits	(1000, 16, 10)
satimage	(1000, 36, 6)	segment	(1000, 19, 7)	stvehicle	(846, 18, 4)
USPS	(1000, 60, 10)	vowel	(990, 10, 11)		
scene13	(1000, 90,13)	cifar10	(1000, 57,10)		

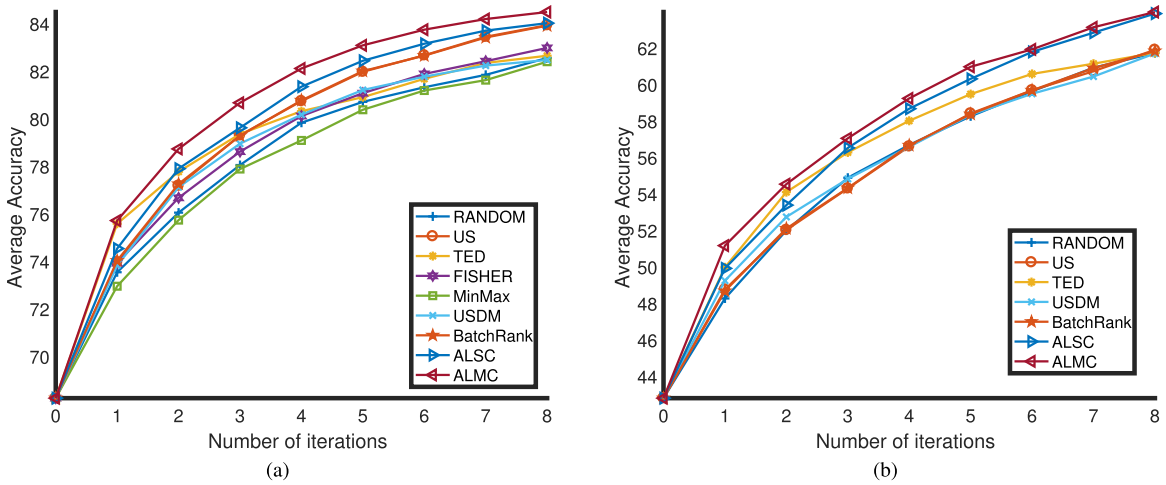


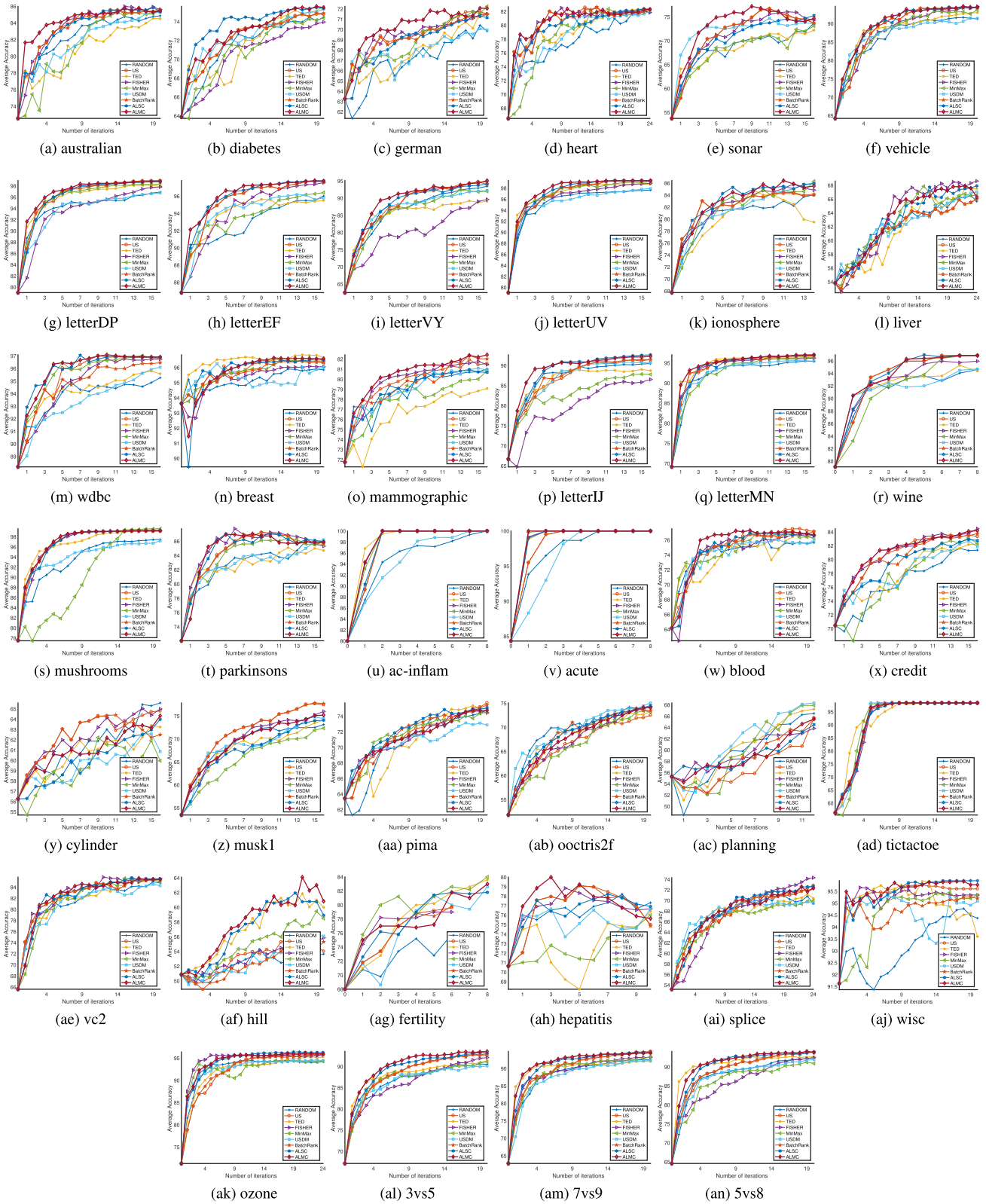
FIGURE 2. The average accuracy of the first eight rounds of batch selection. (a) Shows the average performance over all the binary datasets; (b) shows the average performance over all the multi-class datasets.

of ALC values between ALMC and ALSC over all the binary datasets. It is clear that ALSC perform worse than our method on most of the test datasets. We also test another alternative of ALSC: clustering all the unlabeled examples, instead of clustering only the $m = 3k$ top-ranking unlabeled instances. This approach is still worse than our method, obtaining an average ALC of 0.827. The win/tie/loss counts of ALMC against this approach is 21/16/3, which shows the effectiveness of the proposed method. This indicates that the proposed multi-set clustering is superior to this kind of single-set clustering. The reason is that clustering on multiple sets, which is followed by utilizing TED to select an optimal batch, is more likely to query valuable samples.

Note that we can observe one phenomenon that a small difference of ALC between two active learning methods can still indicate a large gap of the number of annotations required to reach the same accuracy. For example, on the dataset letterUV, ALMC obtains an ALC of 0.976 while Fisher achieves a score of 0.963. The difference between these two scores is only 0.013. However, from Figure 3(j), we observe that ALMC only requires about 5 iterations of batch query to reach the accuracy that Fisher needs 15 iterations to achieve. This means that the number of annotations on the letterUV dataset can be reduced by 67% compared with the Fisher method.

Similarly, our method shows the best performance on multi-class datasets. It performs the best on 9 datasets from 12 multi-class datasets. In Figure 4, we can see that our method never perform worse than random sampling while some other methods may obtain worse performance than random baseline. For example, Batchrank is surpassed by random sampling on satimage, vowel and scene13. All in all, Figure 2 shows the average performance of the first eight iterations of batch selection on both binary and multi-class datasets. We observe that ALMC consistently outperforms the other batch mode active learning approaches.

TED performs well on eight binary datasets, but is far worse than ALMC. The reason is that TED fails to consider the informativeness since it does not utilize label information. Our method, on the contrary, can handle this by employing



(a) australian (b) diabetes (c) german (d) heart (e) sonar (f) vehicle

(g) letterDP (h) letterEF (i) letterVY (j) letterUV (k) ionosphere (l) liver

(m) wdbc (n) breast (o) mammographic (p) letterIJ (q) letterMN (r) wine

(s) mushrooms (t) parkinsons (u) ac-inflam (v) acute (w) blood (x) credit

(y) cylinder (z) musk1 (aa) pima (ab) ooctris2f (ac) planning (ad) tictactoe

(ae) vc2 (af) hill (ag) fertility (ah) hepatitis (ai) splice (aj) wisc

(ak) ozone (al) 3vs5 (am) 7vs9 (an) 5vs8

FIGURE 3. Plots of the average accuracy *w.r.t* the number of batch selection on binary datasets.

the myopic active learning MVAL. US and BatchRank perform similarly to each other. MinMax and USDM obtain slightly better performances than random sampling. The main reason why these methods do not get a very competitively performance is that their performance is very sensitive to the choice of trade-off parameters used in their frameworks.

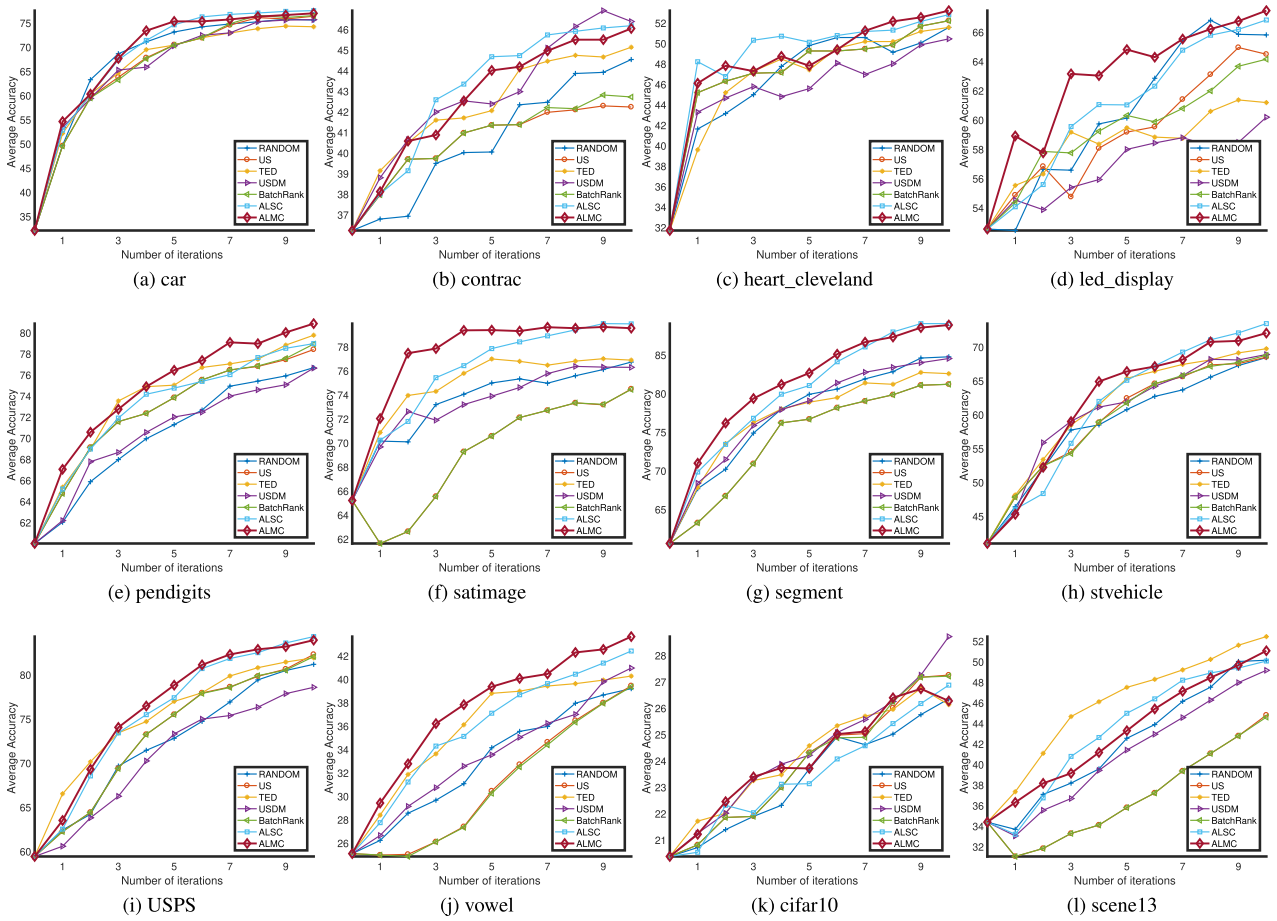


FIGURE 4. Plots of the average accuracy *w.r.t* the number of batch selection on multi-class datasets.

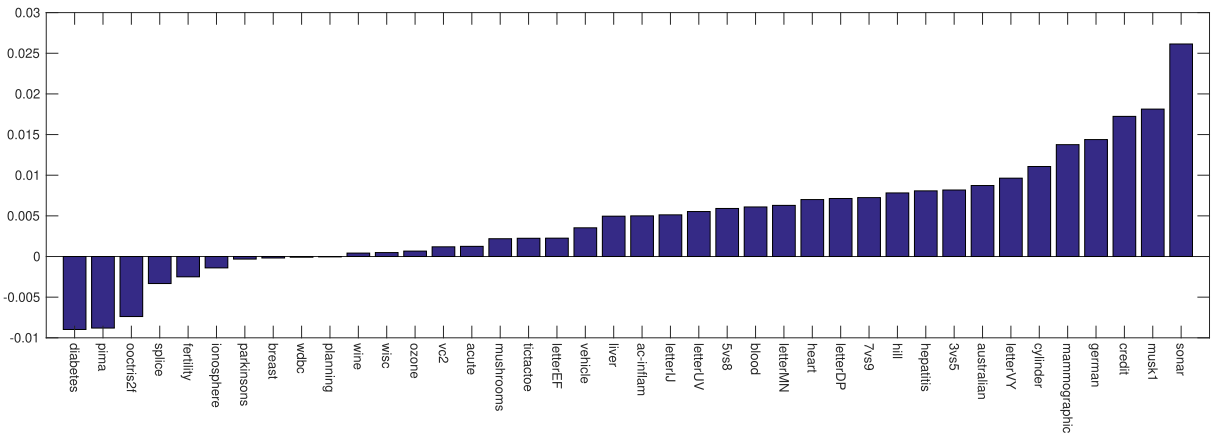


FIGURE 5. Plot of the difference of ALC values between ALMC and ALSC over all the binary datasets. The x-axis is the value of ALC score of ALMC minus that of ALSC.

This parameter should be tuned anew for every dataset, which is hard to do accurately, given the inherent lack of labeled data.

C. THE INFLUENCE OF MYOPIC ACTIVE LEARNER

The myopic active learner in our method takes care of the informativeness of the batch. We illustrate how a choice different from MVAL affects the performance. For this, we use uncertainty sampling (US for short) [20] and expected error reduction (EER) [28] to replace MVAL, resulting in ALMC_US and ALMC_EER, respectively. The experiments results on binary datasets are shown in Table 4. We also demonstrate the plots of the difference of ALC values between ALMC and ALMC_US over all the binary datasets in Figure 6. Similar results of ALMC against ALMC_EER is presented in Figure 7.

TABLE 2. Comparisons of batch mode active learning algorithms in terms of ALC on binary datasets. The algorithm which obtains the best performance or comparable performance is highlighted in bold face and surrounded by a box.

Datasets	RS	US	TED	Fisher	MinMax	USDM	BatchRank	ALSC	ALMC
hill	0.522	0.529	0.571	0.529	0.547	0.527	0.521	0.571	**0.579**
planning	0.595	0.566	0.601	**0.595**	**0.597**	**0.613**	0.582	0.588	0.588
cylinder	0.614	**0.626**	0.606	**0.622**	0.593	0.596	**0.623**	0.600	0.611
liver	0.623	0.615	0.616	**0.635**	**0.628**	0.620	0.615	**0.628**	**0.633**
german	0.670	0.694	0.678	0.694	0.689	0.680	0.692	0.688	**0.703**
splice	**0.676**	**0.676**	0.668	0.671	0.664	**0.676**	**0.676**	**0.680**	0.677
sonar	0.681	0.715	0.683	0.713	0.679	**0.720**	0.715	0.704	**0.731**
musk1	0.689	**0.714**	0.679	0.695	0.656	0.684	**0.715**	0.676	0.694
ooctris2f	**0.693**	0.665	0.676	0.671	0.660	**0.695**	0.682	0.685	0.678
pima	0.706	0.714	0.707	0.713	0.715	0.706	0.717	**0.722**	0.714
diabetes	0.715	0.722	0.716	0.704	0.710	0.723	0.722	**0.738**	0.729
fertility	0.723	0.757	**0.761**	0.769	**0.787**	**0.751**	0.756	**0.767**	0.763
blood	0.743	**0.750**	0.739	0.744	**0.748**	0.742	0.749	0.748	**0.754**
hepatitis	**0.764**	**0.769**	0.729	**0.769**	0.726	0.752	**0.769**	**0.766**	**0.775**
credit	0.768	0.806	0.781	**0.809**	0.773	0.784	0.802	0.794	**0.811**
heart	0.777	**0.805**	0.792	**0.803**	0.785	0.786	**0.804**	0.799	**0.806**
mammographic	0.790	0.796	0.765	0.798	0.780	0.789	0.794	0.791	**0.804**
ionosphere	0.803	0.820	0.801	0.820	0.815	0.813	0.820	**0.828**	**0.827**
vc2	0.815	**0.825**	0.816	**0.826**	0.817	0.811	**0.825**	**0.823**	**0.824**
australian	0.824	0.834	0.809	**0.835**	0.813	0.818	0.834	0.834	**0.843**
parkinsons	0.828	0.846	0.827	**0.853**	0.840	0.828	0.846	**0.848**	**0.848**
letterVY	0.863	0.876	0.852	0.800	0.858	0.867	0.879	0.881	**0.891**
3vs5	0.865	0.881	0.881	0.861	0.869	0.864	0.882	0.893	**0.901**
vehicle	0.867	0.888	0.882	0.879	0.877	0.870	0.888	**0.896**	**0.900**
letterIJ	0.870	0.868	0.863	0.799	0.831	0.869	0.871	**0.886**	**0.891**
5vs8	0.877	0.890	**0.907**	0.859	0.862	0.875	0.892	0.903	**0.909**
7vs9	0.883	0.898	0.899	0.890	0.888	0.874	0.899	0.909	**0.916**
wine	0.915	**0.935**	0.922	0.927	0.917	0.916	**0.935**	**0.937**	**0.937**
tictactoe	**0.919**	**0.920**	**0.926**	**0.917**	**0.920**	**0.920**	**0.920**	**0.919**	**0.921**
letterMN	0.922	**0.941**	**0.945**	0.938	0.925	0.926	0.940	**0.940**	**0.947**
ozone	0.926	0.923	0.921	**0.945**	0.925	0.926	0.923	**0.941**	**0.942**
letterEF	0.929	0.954	0.939	0.943	0.937	0.941	0.954	**0.958**	**0.960**
wisc	0.932	0.954	0.951	0.952	0.945	0.946	0.947	**0.955**	**0.955**
mushrooms	0.935	**0.968**	**0.966**	**0.967**	0.902	0.940	**0.968**	0.967	**0.970**
wdbc	0.937	**0.954**	0.942	0.948	0.956	0.934	0.948	**0.959**	**0.959**
letterDP	0.941	**0.962**	0.959	0.935	0.959	0.936	**0.963**	**0.961**	**0.968**
ac-inflam	0.950	**0.974**	**0.983**	**0.980**	**0.979**	0.948	**0.974**	**0.975**	**0.980**
breast	0.951	0.958	**0.964**	0.953	0.957	0.953	0.957	0.957	0.957
letterUV	0.954	0.969	0.969	0.963	0.968	0.954	0.969	0.970	**0.976**
acute	**0.975**	**0.984**	**0.989**	**0.989**	**0.990**	**0.965**	**0.984**	**0.989**	**0.990**
Average ALC	0.811	0.823	0.817	0.818	0.812	0.813	0.824	0.827	**0.832**
Average Ranking	7.17	4.15	5.88	4.95	6.33	6.55	4.40	3.58	**2.00**
Win Counts	5	15	8	14	7	7	12	21	**31**

We observe that ALMC using MVAL performs better than that using US and EER. ALMC significantly outperforms ALMC_US and ALMC_EER. "W/T/L Counts" also shows the excellent performance of ALMC using MVAL. We conclude that the better the myopic active learner, the better the performance of our method.

Both ALMC_US and ALMC_EER outperform all the competitors except ALSC and ALMC in terms of ALC score. This means that our method is robust and can perform well with other myopic active learning algorithms. We compare ALMC_US and the "top k" version of US and find that ALMC_US significantly exceeds US.

TABLE 3. Comparisons of batch mode active learning algorithms in terms of ALC on multi-class datasets. The algorithm which obtains the best performance or comparable performance is highlighted in bold face and surrounded by a box.

Datasets	RS	US	TED	USDM	BatchRank	ALSC	ALMC
cifar10	0.279	**0.285**	0.275	**0.284**	**0.285**	0.276	0.278
contrac	0.431	0.424	0.430	**0.443**	0.426	**0.445**	**0.443**
scene13	0.469	0.447	**0.492**	0.457	0.447	0.470	0.470
vowel	0.479	0.465	0.457	0.488	0.465	0.501	**0.511**
heart_cleveland	0.507	0.512	0.501	0.508	0.512	**0.525**	**0.526**
stvehicle	0.627	0.629	0.636	0.635	0.627	**0.657**	**0.655**
led_display	0.640	0.637	0.596	0.614	0.638	0.650	**0.656**
car	0.719	0.720	0.696	0.723	0.720	**0.736**	0.728
USPS	0.783	0.797	0.793	0.763	0.796	0.805	**0.811**
satimage	0.786	0.771	0.783	0.782	0.771	0.800	**0.811**
pendigits	0.793	0.839	0.829	0.805	0.838	0.834	**0.844**
segment	0.830	0.829	0.808	0.832	0.829	0.850	**0.855**
Average ALC	0.612	0.613	0.608	0.611	0.613	0.629	**0.632**
Average Ranking	4.83	4.50	5.42	4.33	4.83	2.33	**1.75**
Win Counts	0	1	1	2	1	4	**9**

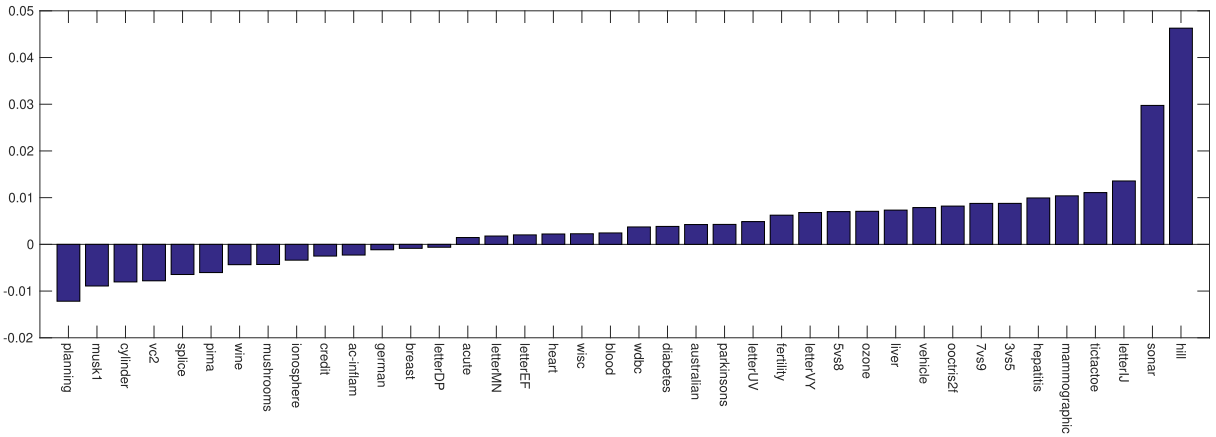


FIGURE 6. Plot of the difference of ALC values between ALMC and ALMC_US over all the binary datasets. The x-axis is the value of ALC score of ALMC minus that of ALMC_US.

TABLE 4. Performance comparisons of using different myopic active learners on binary datasets. "W/T/L Counts" shows the win/tie/loss counts of the proposed method ALMC versus other algorithms over all the datasets, based on the paired t-test at the 95% significance level.

Methods	ALMC_US	ALMC_EER	ALMC
Average ALC	0.828	0.827	**0.832**
W/T/L Counts	18/15/7	17/21/2	-

TABLE 5. Performance comparisons of candidate selection within cluster on binary datasets. "W/T/L Counts" shows the win/tie/loss counts of the proposed method ALMC versus other algorithms over all the datasets, based on the paired t-test at the 95% significance level.

Methods	Rand	Centroid	ALMC
Average ALC	0.828	0.829	**0.832**
W/T/L Counts	15/21/4	15/19/6	-

Similarly, the "top k" version of EER only achieves an average ALC of 0.817 while ALMC_EER obtains a better result 0.827. And the win/tie/loss counts of ALMC_EER against EER is 24/12/4. This indicates that our method can successfully adapt a myopic active learner to a batch version. Even when the myopic learner is random sampling (ALMC_RS for short), we can still observe an improvement in the performance. ALMC_RS exceeds random sampling with an average ALC of 0.817. And the win/tie/loss counts of ALMC_RS versus random sampling is 18/16/6, which verifies the effectiveness of the proposed framework. Figure 8 shows the difference of the performance of ALMC_RS against random sampling on all the binary test sets.

D. THE INFLUENCE OF RANKING ORDER

We also investigate the effect of the ranking order provided by myopic learner. We randomize the ranking order but still use MVAL to select the most informative candidate from each cluster. In other words, the examples used by our multi-set clustering are not selected according to the myopic

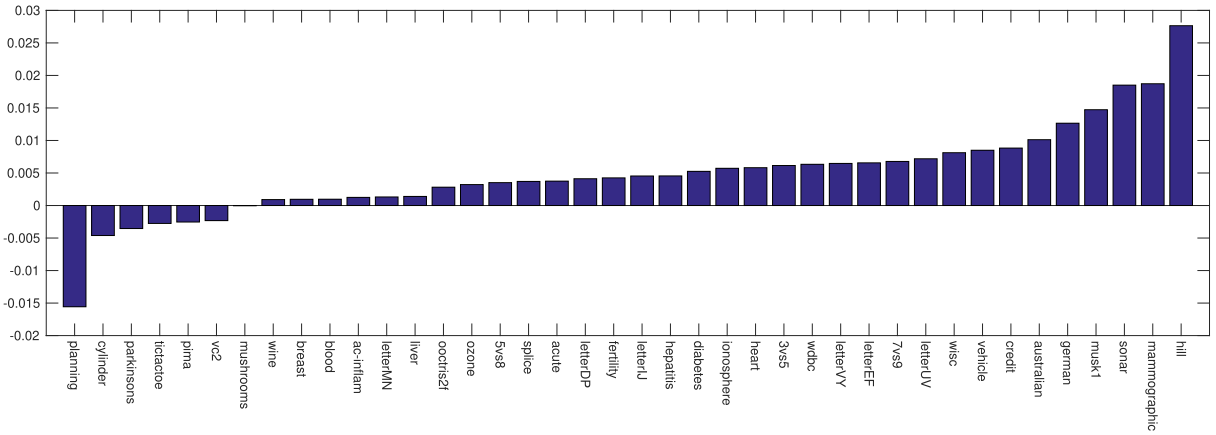


FIGURE 7. Plot of the difference of ALC values between ALMC and ALMC_EER over all the binary datasets. The x-axis is the value of ALC score of ALMC minus that of ALMC_EER.

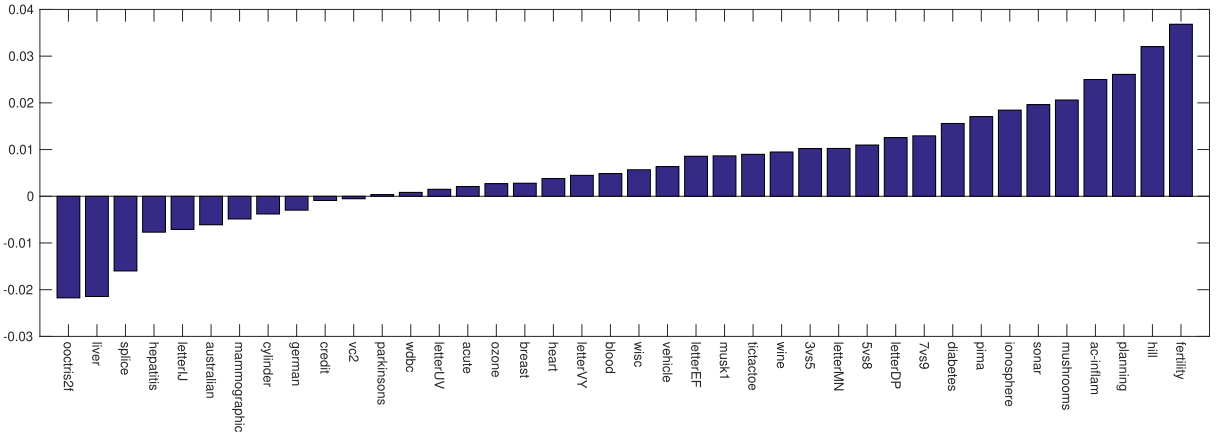


FIGURE 8. Plot of the difference of ALC values between ALMC_RS and random sampling over all the binary datasets. The x-axis is the value of ALC score of ALMC_RS minus that of random sampling.

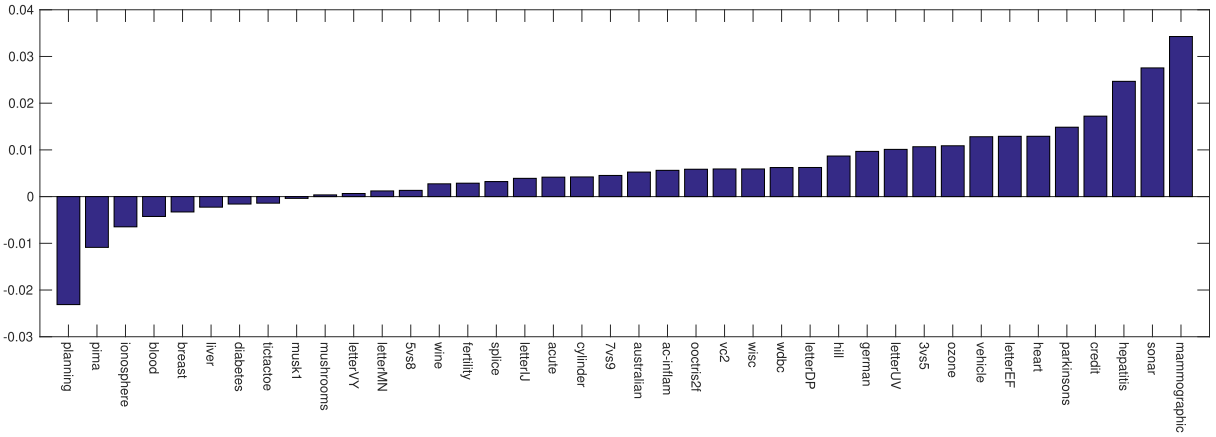


FIGURE 9. Plot of the difference of ALC values between ALMC and ALMC_RR over all the binary datasets. The x-axis is the value of ALC score of ALMC minus that of ALMC_RR.

active learning algorithm any more. Instead, they are randomly chose and utilized for clustering. We call this approach ALMC_RR for short.

We find that this approach obtains an average ALC of 0.826 on binary datasets, which is worse than our method. In addition, the win/tie/loss counts shows that our method exceeds it on 19 datasets and only fails on 3 datasets. We also show the difference of the performance of ALMC against ALMC_RR on all the binary test sets in Figure 9. Clearly, ALMC performs much better than ALCM_RR on most datasets. This indicates that the ranking order provided by the myopic active learner is useful.

E. SELECTION WITHIN CLUSTER

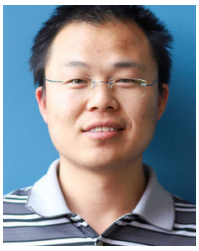
In each cluster, we choose the sample which ranks first according to the myopic active learner. Here we consider two alternatives: selecting an instance uniformly and randomly (Rand for short) and choosing an instance which lies closest to the centroid (Centroid for short). MVAL is utilized as the myopic active learner. As shown in Table 5, we see that querying the instance with highest ranking outperforms the other two choices in terms of both the average ALC and the win/tie/loss counts. The reason is that such a selection will preserve the informativeness as much as possible.

V. CONCLUSION

In this paper, we proposed a novel batch model active learning method called ALMC. ALMC employs a myopic active learner to rank the unlabeled samples. Then, according to the ranking order, ALMC conducts clustering to multiple informative sets. Finally, the proposed method chooses an optimal batch by minimizing the expected predictive variance. In particular, in our method, the myopic active learner is utilized to maintain the informativeness while the clustering algorithm is used to keep the diversity of selected samples. The subsequent batch selection promotes the representativeness of selected samples. Extensive experiments on 52 benchmark datasets (both binary and multi-class datasets) demonstrate that the proposed algorithm outperforms current state-of-the-art batch mode active learning methods.

REFERENCES

[1] E. Biyik, K. Wang, N. Anari, and D. Sadigh, "Batch active learning using determinantal point processes," 2019, *arXiv:1906.07975*. [Online]. Available: http://arxiv.org/abs/1906.07975
[2] K. Brinker, "Incorporating diversity in active learning with support vector machines," in *Proc. 20th Int. Conf. Mach. Learn.*, 2003, pp. 59–66.
[3] S. Chakraborty, V. Balasubramanian, Q. Sun, S. Panchanathan, and J. Ye, "Active batch selection via convex relaxations with guaranteed solution bounds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 10, pp. 1945–1958, Oct. 2015.
[4] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, Apr. 2011.
[5] R. Chattopadhyay, Z. Wang, W. Fan, I. Davidson, S. Panchanathan, and J. Ye, "Batch mode active sampling based on marginal probability distribution matching," *ACM Trans. Knowl. Discovery Data*, vol. 7, no. 3, pp. 1–25, Sep. 2013.
[6] Y. Chen and A. Krause, "Near-optimal batch mode active learning and adaptive submodular optimization," in *Proc. The 30th Int. Conf. Mach. Learn.*, 2013, pp. 160–168.
[7] D. J. Cook and N. C. Krishnan, *Activity Learning: Discovering, Recognizing, and Predicting Human Behavior From Sensor Data*. Hoboken, NJ, USA: Wiley, 2015.
[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 886–893.
[9] B. Demir, C. Persello, and L. Bruzzone, "Batch-mode active-learning methods for the interactive classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 3, pp. 1014–1031, Mar. 2011.
[10] B. Du, Z. Wang, L. Zhang, L. Zhang, W. Liu, J. Shen, and D. Tao, "Exploring representativeness and informativeness for active learning," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 14–26, Jan. 2017.
[11] F.-F. Li and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR05)*, Jun. 2005, pp. 524–531.
[12] A. Freytag, E. Rodner, and J. Denzler, "Selecting influential examples: Active learning with expected model output changes," in *Proc. ECCV*. Zürich, Switzerland: Springer, 2014, pp. 562–577.
[13] Y. Guo, "Active instance sampling via matrix partition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 802–810.
[14] Y. Guo and D. Schuurmans, "Discriminative batch mode active learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 593–600.
[15] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Batch mode active learning and its application to medical image classification," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 417–424.
[16] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Semi-supervised SVM batch mode active learning for image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–7.
[17] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, May 1994.
[18] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.
[19] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Dec. 1998.
[20] D. D. Lewis and W. A. Gale, "A sequential algorithm for training text classifiers," in *Proc. 17th ACM SIGIR*, 1994, pp. 3–12.
[21] M. Lichman. (2013). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml
[22] R. Luo and X. Wang, "Batch active learning with two-stage sampling," *IEEE Access*, vol. 8, pp. 46518–46528, 2020.
[23] R. Mehrotra and E. Yilmaz, "Representative & informative Query selection for learning to rank using submodular functions," in *Proc. 38th ACM SIGIR*, 2015, pp. 545–554.
[24] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.
[25] S. Patra and L. Bruzzone, "A batch-mode active learning technique based on multiple uncertainty for SVM classifier," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 3, pp. 497–501, May 2012.
[26] S. Patra and L. Bruzzone, "A cluster-assumption based batch mode active learning technique," *Pattern Recognit. Lett.*, vol. 33, no. 9, pp. 1042–1048, Jul. 2012.
[27] R. Pinsler, J. Gordon, E. Nalisnick, and J. M. Hernández-Lobato, "Bayesian batch active learning as sparse subset approximation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 6359–6370.
[28] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. 18th ICML*, 2001, pp. 441–448.
[29] A. I. Schein and L. H. Ungar, "Active learning for logistic regression: An evaluation," *Mach. Learn.*, vol. 68, no. 3, pp. 235–265, Aug. 2007.
[30] B. Settles, "Active learning literature survey," *Univ. Wisconsin*, vol. 52, nos. 55–66, p. 11, 2010.
[31] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, 1992, pp. 287–294.
[32] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *J. Mach. Learn. Res.*, vol. 2, pp. 45–66, Nov. 2002.
[33] H. Wang, R. Zhou, and Y.-D. Shen, "Bounding uncertainty for active batch selection," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 5240–5247.
[34] Z. Wang, B. Du, W. Tu, L. Zhang, and D. Tao, "Incorporating distribution matching into uncertainty for multiple kernel active learning," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 1, pp. 128–142, Jan. 2019.
[35] Z. Wang and J. Ye, "Querying discriminative and representative samples for batch mode active learning," *ACM Trans. Knowl. Discovery Data*, vol. 9, no. 3, pp. 1–23, Apr. 2015.
[36] Y. Yang and M. Loog, "A benchmark and comparison of active learning for logistic regression," *Pattern Recognit.*, vol. 83, pp. 401–415, Nov. 2018.
[37] Y. Yang, Z. Ma, F. Nie, X. Chang, and A. G. Hauptmann, "Multi-class active learning by uncertainty sampling with diversity maximization," *Int. J. Comput. Vis.*, vol. 113, no. 2, pp. 113–127, Jun. 2015.
[38] D. Yoo and I. S. Kweon, "Learning loss for active learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 93–102.
[39] K. Yu, J. Bi, and V. Tresp, "Active learning via transductive experimental design," in *Proc. 23rd Int. Conf. Mach. Learn. (ICML)*, 2006, pp. 1081–1088.
[40] B. Zhang, L. Li, S. Yang, S. Wang, Z.-J. Zha, and Q. Huang, "State-relabeling adversarial active learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8756–8765.



YAZHOU YANG received the B.S. degree in information system engineering and the M.S. degree in control science and engineering from the National University of Defense Technology, Changsha, China, in 2011 and 2013, respectively, and the Ph.D. degree from the Pattern Recognition Laboratory, Delft University of Technology, Delft, The Netherlands, in 2018. He is currently a Research Assistant with the National University of Defense Technology. His current research interests include active learning, semi-supervised learning, deep learning, and image classification.

XIAOQING YIN received the B.S. and Ph.D. degrees in system engineering from the National University of Defense Technology, Changsha, China, in 2011 and 2018, respectively. He is currently a Lecturer with the Department of Advanced Interdisciplinary, National University of Defense Technology. His research interests include computer vision and image processing.

YANG ZHAO received the Ph.D. degree from the College of Systems Engineering, National University of Defense Technology, in 2009. In 2009, he was employed as a Tutor and an Associate Professor with the National University of Defense Technology. His research interests include resource scheduling, task planning, and intelligent computing.

JUN LEI received the B.S. degree in information system engineering and the M.S. and Ph.D. degrees in control science and engineering from the National University of Defense Technology, Changsha, China, in 2010, 2012, and 2017, respectively. He is currently a Lecturer with the National University of Defense Technology. His current research interests include machine learning, computer vision, and deep learning.

WEILI LI received the B.S. and Ph.D. degrees in system engineering from the National University of Defense Technology, Changsha, China, in 2012 and 2019, respectively. She is currently a Lecturer with the Department of System Engineering, National University of Defense Technology. Her current research interests include computational photography and image deblurring.

ZHE SHU received the B.S. degree from Wuhan University, Wuhan, China, in 2011, and the Ph.D. degree in management science and engineering from the National University of Defense Technology (NUDT), Changsha, China, in 2019. He is currently an Assistant Research Fellow with the National University of Defense Technology. His current research interests include evolutionary computation, multi objective optimization, and system of systems architecture modeling and optimization.