Hindawi
Computational Intelligence and Neuroscience
Volume 2020, Article ID 3287589, 13 pages
https://doi.org/10.1155/2020/3287589

Research Article

Double-Criteria Active Learning for Multiclass Brain-Computer Interfaces

Qingshan She,[1] **Kang Chen**,[1] **Zhizeng Luo**,[1] **Thinh Nguyen**,[2] **Thomas Potter**,[2] **and Yingchun Zhang**[2]

[1]*Institute of Intelligent Control and Robotics, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China*
[2]*Department of Biomedical Engineering, University of Houston, Houston, TX 77204, USA*

Correspondence should be addressed to Qingshan She; qsshe@hdu.edu.cn and Yingchun Zhang; yzhang94@uh.edu

Received 25 July 2019; Accepted 11 February 2020

Guest Editor: Eduardo Rodriguez-Tello

Copyright © 2020 Qingshan She et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recent technological advances have enabled researchers to collect large amounts of electroencephalography (EEG) signals in labeled and unlabeled datasets. It is expensive and time consuming to collect labeled EEG data for use in brain-computer interface (BCI) systems, however. In this paper, a novel active learning method is proposed to minimize the amount of labeled, subject-specific EEG data required for effective classifier training, by combining measures of uncertainty and representativeness within an extreme learning machine (ELM). Following this approach, an ELM classifier was first used to select a relatively large batch of unlabeled examples, whose uncertainty was measured through the best-versus-second-best (BvSB) strategy. The diversity of each sample was then measured between the limited labeled training data and previously selected unlabeled samples, and similarity is measured among the previously selected samples. Finally, a tradeoff parameter is introduced to control the balance between informative and representative samples, and these samples are then used to construct a powerful ELM classifier. Extensive experiments were conducted using benchmark and multiclass motor imagery EEG datasets to evaluate the efficacy of the proposed method. Experimental results show that the performance of the new algorithm exceeds or matches those of several state-of-the-art active learning algorithms. It is thereby shown that the proposed method improves classifier performance and reduces the need for training samples in BCI applications.

1. Introduction

Brain-computer interfaces (BCIs) are systems that allow users to control external devices via observed brain activity, without relying on peripheral nerve or muscle activity [1]. The most common and useful BCIs are constructed using noninvasive brain activity recording techniques, such as electroencephalography (EEG) [2]. While EEG has become widely used for medical monitoring, rehabilitation, neuroprosthesis, and other healthcare applications [3–5], the data acquisition process can be lengthy and exhaustive for users [6]. In addition, EEG signals often vary over the course of an experiment due to both biological and technical causes, including subject-specific anatomical differences, intersession variability, and the attentional drift of subjects [7]. Consequently, users must often undergo a long data collection process to train a suitable BCI system. This poses a prohibitive burden for individuals with paralysis or a severely injured central nervous system, making it a major hurdle for therapeutic applications. It is therefore of the utmost importance that developed BCI systems achieve efficient and robust performance with as few samples as possible.

One approach that has been effectively applied to cases with limited training sets is the introduction of active learning (AL) to the BCI calibration procedure. AL queries the class labels of informative samples within the unlabeled sample space to maximize the efficiency of the learning model, and its application greatly reduces the complexity of training samples without any obvious loss of classification accuracy [8]. In essence, AL is an iterative sampling and labeling procedure. On each iteration, AL extracts the

sample or batch of samples that are most valuable for improving the current classification model from the unlabeled data pool, and these samples are then manually labeled. The greatest challenge for AL methods is identifying the most informative samples so that the maximum prediction accuracy can be achieved. A number of sample-selection criteria have then been applied to this task, including (1) query-by-committee (QBC), in which several distinct classifiers are used and the selected samples are those with the largest difference between the labels predicted by different classifiers [9–11]; (2) margin uncertainty sampling, wherein the samples are selected according to the maximum uncertainty based on their respective distances from the classification boundaries [12, 13]; (3) max-entropy sampling, which uses entropy as the uncertainty measure via probabilistic modeling [14, 15]; and (4) diversity sampling, which prefers selecting representative samples [16].

Over the past few decades, many supervised learning models have been adopted as baseline classifiers for AL, including linear discriminate analysis (LDA) [12, 17], support vector machine (SVM) [18, 19], artificial neural network (ANN) [20], and extreme learning machine (ELM) [21, 22]. Among these, the ELM has shown a high learning speed and good generalizability in preliminary testing. Additionally, it can be directly applied to both two-class and multiclass classification. To date, few studies have attempted to introduce AL algorithms into the ELM framework, although these have shown the method to be competitive with active SVMs [13, 14, 23]. Specifically, Yu et al. [13] proposed an active learning method called AL-ELM with the goal of saving training time, and results showed a classification performance comparable to that of AL-SVM [18]. Zhang and Er [23] then introduced the SEAL-ELM method by combining the online sequential ELM (OS-ELM) with AL, yielding a higher classification accuracy than offline combinations of AL and SVM on most test datasets. Regrettably, these existing active ELMs only consider a single-querying strategy, leaving space for improvement. The intuitive next step was to then introduce multiple querying strategies to select desirable samples. In fact, researchers have tried to combine two strategies in AL with base classifier SVM, with each performing better than their single-query counterparts [24–26]. At present, however, few implementations of active learning with ELM have been explored and applied for motor imagery- (MI-) based BCI systems [8, 13].

The present investigation intends to fill this gap by combining a two-query AL algorithm with an ELM and testing the method in a BCI application. A well-defined, general framework for active learning is thereby developed in a manner that accounts for both informativeness and representativeness in a multiclass situation. First, an uncertainty sampling strategy is adopted to select a relative large number of samples using the base ELM classifier. The degree of diversity between labeled training data and previously selected, unlabeled samples is then assessed, along with the degree of similarity between the unlabeled samples. Finally, highly informative and representative samples are used to update the ELM classifier through the introduction of a tradeoff parameter. The method is then tested on several benchmark datasets, along with a multiclass MI EEG dataset from BCI Competition IV Dataset 2a. Results demonstrate that the performance of the new method compares favorably with that of existing AL approaches.

Compared to existing ELM-based active learning algorithms, the new method has several noteworthy aspects:

(1) Considering that the use of a single uncertainty strategy may not take full advantage of the abundant information with unlabeled data, the AL-ELM algorithm is extended to combine two querying strategies (uncertainty and diversity) in order to select the most valuable samples from the unlabeled EEG data pool.

(2) The proposed algorithm provides a straightforward and meaningful way to measure representativeness by assaying two kinds of similarity: the similarity between a query sample and the labeled dataset, and the similarity between any two possible query samples. Employing this modified diversity strategy can help isolate highly representative samples during the active learning process.

2. Background Knowledge

2.1. Active Learning. Active learning methods typically comprise five basic components: **L**, **U**, **T**, **Q**, and **S**. **L** is the limited labeled dataset, **U** is the pool of samples/instances that contains abundant unlabeled instances, **T** is the classification model trained by **L**, **Q** is a query strategy to select the most valuable instances from **U**, and **S** is a human annotator that labels the selected instances correctly. AL is an iterative procedure that gradually adds the most important samples, queried by **Q** and labeled by **S**, from **U** to **L** to update the classification model **T**. The iterative AL process will continue in this manner until a predefined criterion is met. The ability to identify both an excellent classification model **T** and an effective query strategy **Q** is highly important for active learning algorithms.

Depending on the number of querying samples at each iteration, AL can be divided into two groups: stream-based AL and pool-based AL. In stream-based AL, the learner can only access one sample per iteration, while pool-based AL allows the learner to select a batch of samples during each iteration. Adjusting the selection method and number of queried samples then creates different AL algorithms, such as the QBC strategy, the uncertainty strategy, and the diversity strategy.

2.2. Basic ELM. Single-hidden-layer feedforward neural networks (SLFNs) are capable of universal approximation [21]. Consider a dataset containing N training samples, $\{X, Y\} = \{x_j, y_j\}_{j=1}^{N}$, with the input $x_j = [x_{j1}, x_{j2}, \ldots, x_{jp}]^T \in \mathbb{R}^p$ and a corresponding desired output of $y_j = [y_{j1}, y_{j2}, \ldots, y_{jq}]^T \in \mathbb{R}^q$, where p and q represent the respective dimensions and T denotes a transpose operation. Assuming that M is the number of hidden neurons, the output function of the SLFNs is mathematically modeled as

$$\mathbf{y}_j = \sum_{i=1}^{M} \boldsymbol{\beta}_i g\left(\mathbf{a}_i^{\mathrm{T}} \mathbf{x}_j + b_i\right), \quad j = 1, \ldots, N, \tag{1}$$

where $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{iq}]^{\mathrm{T}} \in \mathbb{R}^q$ is the weight vector that connects the i-th hidden neuron to the output neurons, $\mathbf{a}_i = [a_{i1}, a_{i2}, \ldots, a_{ip}]^{\mathrm{T}} \in \mathbb{R}^p$ is a randomly chosen input weight vector connecting the i-th hidden neuron to the input neurons, $b_i \in \mathbb{R}\,(i = 1, \ldots, M)$ is a randomly chosen bias of the i-th hidden node, and $g(\bullet)$ is the activation function, which can be any nonlinear piecewise continuous function (such as a sigmoid function or Gaussian function).

For convenience, equation (1) can be rewritten in matrix notation as

$$\mathbf{Y} = \mathbf{H}\boldsymbol{\beta}, \tag{2}$$

where $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N]^{\mathrm{T}} \in \mathbb{R}^{N \times q}$ is the expected network output, $\boldsymbol{\beta} = [\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_M]^{\mathrm{T}} \in \mathbb{R}^{M \times q}$ denotes the weight of output layer, and $\mathbf{H}$ is the hidden layer output matrix which is defined as

$$\mathbf{H} = \begin{bmatrix} g\left(\mathbf{a}_1^{\mathrm{T}} \mathbf{x}_1 + b_1\right) & \cdots & g\left(\mathbf{a}_M^{\mathrm{T}} \mathbf{x}_1 + b_M\right) \\ \cdots & \cdots & \cdots \\ g\left(\mathbf{a}_1^{\mathrm{T}} \mathbf{x}_N + b_1\right) & \cdots & g\left(\mathbf{a}_M^{\mathrm{T}} \mathbf{x}_N + b_M\right) \end{bmatrix}_{N \times M}. \tag{3}$$

Unlike SLFNs, which require that the parameters of hidden neurons are adjusted during training, ELM adopts randomly generated hidden layer parameters and a tuning-free training strategy [22]. Even with these random hidden node parameters, ELM maintains the universal approximation capability of SLFNs [21]. The ELM training then aims to find suitable network parameters to minimize the approximation error $\|\mathbf{H}\boldsymbol{\beta} - \mathbf{Y}\|_2$. To achieve better generalization performance, a regularization parameter c is introduced in [27], with its corresponding objective function given as

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|_2^2 + \frac{c}{2}\|\mathbf{H}\boldsymbol{\beta} - \mathbf{Y}\|_2^2, \tag{4}$$

where $\|\cdot\|_2$ denotes the l_2-norm of a matrix or a vector. We can obtain the output weight vector $\boldsymbol{\beta}$ using the Moore-Penrose principle. The solution of equation (4) is then $\boldsymbol{\beta} = ((\mathbf{I}/c) + \mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{Y}$ if $N > M$, and $\boldsymbol{\beta} = \mathbf{H}^{\mathrm{T}}((\mathbf{I}/c) + \mathbf{H}\mathbf{H}^{\mathrm{T}})^{-1}\mathbf{Y}$ if $N < M$.

3. The D-AL-ELM Method

In this section, we present a novel active learning algorithm, D-AL-ELM, that incorporates both the uncertainty and diversity strategies into consecutive steps. This identifies the most valuable, informative instances, which can then be selected to update the baseline classifier ELM during each learning round.

3.1. Discriminative Information by the Uncertainty Criterion. The uncertainty criterion is used to measure the informativeness of each sample. Uncertain samples which lie along the boundaries of different classes carry more information and play a more significant role in the construction of a classifier. In this implementation, the best-versus-second-best (BvSB) strategy is adopted to estimate the uncertainty of each sample. The BvSB strategy is based on a calculation of posterior probability, which considers the difference in probability values between the two classes with the highest estimated probabilities [28]. The outputs of the ELM then approximate the posterior probabilities of the different classes [13]. To do this, a sigmoid function is used to construct a mapping relationship between the real outputs of the ELM and the posterior probabilities, which is described as

$$p\left(y = 1 | f_i(x)\right) = \frac{1}{1 + \exp\left(-f_i(x)\right)}, \tag{5}$$

where $f_i(x)$ denotes the actual output of the i–th output node corresponding to the time instance x. In practice, equation (5) is only applied to two-class problems, such that the sum of the converted posterior probabilities for the instance x is always 1. However, application in multiclass problem may create a summed posterior proximity that exceeds 1, so calculated probabilities were normalized using the following formula:

$$\overline{p}\left(y = 1 | f_i(x)\right) = \frac{p\left(y = 1 | f_i(x)\right)}{\sum_{j=1}^{q} p\left(y = 1 | f_j(x)\right)}, \tag{6}$$

where $p(y = 1 | f_i(x))$ is the original probability of the i – th class.

Based on the above parameters, the BvSB strategy for each sample x can be expressed as

$$f(\mathbf{x})^{BvSB} = p\left(y_{\text{best}} | x\right) - p\left(y_{\text{second-best}} | x\right), \tag{7}$$

where $p(y_{\text{best}} | x)$ and $p(y_{\text{second-best}} | x)$ are the largest and second largest posterior probabilities of x, respectively. It should be noted that $f(\mathbf{x})^{BvSB}$ values are inversely related to the amount of uncertainty in a sample, with smaller values indicating greater uncertainty.

3.2. Representative Information by the Diversity Criterion. The selection of redundant or overly similar samples is of little use when attempting to construct a robust classifier. It is therefore necessary to use a diversity criterion to select a batch of samples which are diverse in nature. A feasible way of measuring the diversity of uncertain samples is the cosine angle distance. Following this approach, the similarity between two samples x_i and x_j is given by

$$S\left(x_i, x_j\right) = \left|\cos\left(x_i, x_j\right)\right| = \frac{\left|x_i \cdot x_j\right|}{\left\|x_i\right\|\left\|x_j\right\|}. \tag{8}$$

As can be seen from equation (8), the similarity $S(x_i, x_j)$ between the two samples x_i and x_j is small if these two samples are far from each other, and vice versa.

Suppose a batch of samples $\mathbf{W} = \{w_1, w_2, \ldots, w_n\}$. If the value of $max_{i=1,\ldots,n} S(x, w_i)$ is small, then the new sample x is diverse from the samples in $\mathbf{W}$. The similarity between a new sample x and $\mathbf{W}$ is defined as

$$div(x, \mathbf{W}) = \max_{w_j \in W} S\left(\mathrm{x}, \mathrm{w}_j\right). \tag{9}$$

Note that a smaller $div(x, \mathbf{W})$ value implies more diversity between x and $\mathbf{W}$.

In order to avoid selecting highly redundant samples, a novel diversity criterion is defined by combining the similarity between a query sample and the labeled set, and the similarity between any two candidate query samples at the same time. This calculation is given by

$$div(w_i) = div(w_i, \mathbf{W}) + div(w_i, \mathbf{L}), \tag{10}$$

where $div(w_i, \mathbf{W})$ represents the diversity between the sample w_i and the candidate set $\mathbf{W}$ (apart from w_i), and $div(w_i, \mathbf{L})$ represents the diversity between the sample w_i and the labeled training set $\mathbf{L}$.

3.3. Proposed D-AL-ELM Algorithm. The BvSB sampling method is a highly effective strategy for sample selection in active learning. Unfortunately, the BvSB may also select some uncertain samples which contain highly redundant information, which reduces the information available for classification. To address this problem, optimal samples were selected for classification. An ideal sample would not only furnish significant information for the classifier but also show diversity from the candidate unlabeled set and a minimal amount of redundancy within the labeled set.

The specific steps for each iteration of the D-AL-ELM algorithm are as follows:

Step 1: the BvSB strategy is adopted to select the h most uncertain samples from the unlabeled samples pool $\mathbf{U}$.

Step 2: let h represent the most uncertain samples, denoted by $\mathbf{W} = \{w_1, w_2, \ldots, w_h\} \subseteq \mathbf{U}$, and $\mathbf{S}_m$ be an arbitrary subset containing $m (m \leq h)$ samples selected from $\mathbf{W}$. Two evaluations are then performed, including the diversity from the labeled set $\mathbf{L}$ and the candidate set $\mathbf{S}_m$, and the similarity to the samples in $\mathbf{S}_m$.

Step 3: combining the discriminative and representative parts, the following formulation is obtained to select the m samples which are uncertain and diverse from each other:

$$\hat{\mathrm{x}}^{BvSB-div} = \arg\min_{s_i \in \mathbf{S}_m} \{\lambda f(s_i)^{BvSB} + (1-\lambda)(div(s_i, \mathbf{S}_m) + div(s_i, \mathbf{L}))\}, \tag{11}$$

where λ is a tradeoff parameter that can balance the informativeness and representativeness criteria, and $\mathbf{L}$ is the labeled training set. $\hat{\mathbf{x}}^{BvSB-div}$ denotes the unlabeled sample that will be annotated and then included into the labeled training dataset for updating the ELM classifier.

The implementation of the proposed method is summarized in Algorithm 1.

In order to quantitatively evaluate the quality of each learning algorithm, area under the learning curve (ALC) [13] was calculated as a performance metric, which is described as

$$\mathrm{ALC} = \sum_{i=0}^{N_{iter}-1} \frac{y_i + y_{i+1}}{2N_{iter}}, \tag{12}$$

where N_{iter} denotes the number of learning iterations and y_i denotes the classification accuracy at the i-th learning round, such that ALC $\in [0, 1]$. It is noted that the larger the ALC value, the better the performance of the learning algorithm.

4. Experimental Results and Discussions

In this section, several experiments were performed on benchmark datasets and multiclass MI EEG datasets to evaluate the performance of the proposed D-AL-ELM method, in comparison with the other state-of-the-art approaches, including passive learning-based ELM, AL-ELM [13], and entropy-based ELM [14]. All methods were implemented using the MATLAB 2014b environment on a computer with a 2.5 GHz processor and 4.0 GB RAM.

4.1. Experiments on the Benchmark Datasets

4.1.1. Description of the Benchmark Datasets. A series of experiments were performed to evaluate the D-AL-ELM algorithm on 9 benchmark datasets from the KEEL dataset [29] and UCI dataset repositories [30]. Datasets included both binary and multiclass classification problems. As in [13], each raw dataset was divided into three parts: a small initial labeled set, a large unlabeled set, and a testing set. Testing instances comprised 50% of the total number of samples, while the percentage of initially labeled instances was assigned based on the size of the raw dataset and the number of categories. Detailed information regarding these datasets is presented in Table 1.

4.1.2. The Compared Algorithms, Parameter Settings, and the Performance Metric. In our experiments, we compare the proposed method with other state-of-the-art learning algorithms, including the following:

(1) PL-ELM: a passive learning algorithm that randomly selects some instances from the unlabeled set to train the initial classifier

(2) AL-ELM: a batch-mode active learning method based on ELM that uses the margin sampling strategy to select most uncertain examples for labeling [13]

(3) ELM-Entropy: querying discriminative samples through entropy measures [14]

In this study, the ELM adopted a sigmoid function as the activation function on the hidden level. A grid search based on tenfold cross-validation was then used to find the optimal number of hidden nodes M in the initial labeled set. For the regularization parameter c, a leave-one-out (LOO) cross-validation strategy was adopted based on the minimum MSE^{PRES} to find the optimal parameter value [31]. The optimal parameters M and c were determined from $M \in \{10, 20, \ldots, 200\}$ and $c \in \{e^{-5}, e^{-4.9}, \ldots, e^{5}\}$ on all the datasets except for the Letter dataset, where the parameter M was searched among $\{100, 200, \ldots, 1000\}$. Additionally, the tradeoff parameter $\lambda \in \{0.1, 0.2, \ldots, 0.9\}$ for equation (10) was chosen by grid search when M and c were fixed through the aforementioned methods. It should be noted that the

Inputs: $\mathbf{L} = \{(x_i, y_i)\}$ with n_l labeled samples, $\mathbf{U} = \{x_i\}$ with n_u unlabeled samples ($n_u \gg n_l$), the tradeoff parameter (λ), the number of samples selected on basis of their uncertainty (h), the batch size (m), and the terminating condition.
Output: The final learned ELM classifier.
(1) Train the ELM classifier using labeled set **L**.
(2) **Repeat**
(3) Calculate the estimated probability for the samples in **U** with the pretrained ELM classifier according to equation (5) or (6).
(4) Calculate the uncertainty level of each sample in **U** using equation (7).
(5) Include the h most uncertain samples into the set **W**.
(6) Select m samples from **W** using equation (11).
(7) Label the selected m samples.
(8) Update the labeled set **L** and unlabeled set **U**.
(9) Use the extended set **L** to train a new ELM classifier.
(10) **Until** the terminating condition is satisfied.
(11) Return the output the final learned ELM classifier.

Algorithm 1: The double-criteria active learning with the ELM algorithm.

Table 1: Details of the datasets including the numbers of the corresponding features and samples.

Dataset	Number of			Percentage of initial labeled instances (%)	Percentage of initial unlabeled instances (%)	Percentage of test instances (%)
	Features	Instances	Classes			
Liver	7	345	2	10	40	50
Diabetes	8	768	2	10	40	50
Wdbc	30	569	2	10	40	50
Twonorm	20	7400	2	1	49	50
Hayes-Roth	4	160	3	10	40	50
Iris	4	150	3	10	40	50
Wine	13	178	3	10	40	50
Segment	19	2310	7	10	40	50
Letter	16	20000	26	1	49	50

ELM parameter selection process was implemented in the same manner for all four methods.

Parameter details are shown in Table 2. It should be noted that the regularization parameter c was automatically identified using the LOO cross-validation and was not fixed during the learning process (thus, not shown in Table 2).

The batch mode was adopted to add new labeled instances. For the proposed D-AL-ELM method, h samples were first selected from the unlabeled set using equation (7), and then m samples were selected from the h samples using equation (11) and added to the labeled set for each iteration. In this experiment, h was empirically set to $h = 5m$ while m was 5% of the total instances in the original unlabeled set for 8 of the 9 datasets (except Letter). For the Letter dataset, m was 1% of the total instances in the original unlabeled set and h was set to $h = 2m$. These parameters were chosen to decrease the labeling cost, considering the size of the raw dataset and the number of categories.

To provide a fair comparison, all four methods queried m instances on each iteration. For each dataset, the procedure was stopped when the prediction accuracy stabilized or the number of selected samples was greater than 80% of the original unlabeled set. Additionally, to ensure the validity of experimental results, ten runs were performed for each learning method in each experiment, and average results were calculated.

4.1.3. Comparisons with Relevant State-of-the-Art Algorithms. Figure 1shows the trends of classification accuracy for the classifiers when trained by increasing numbers of data points across the various datasets. The results show that the proposed D-AL-ELM algorithm yielded the highest accuracy of all four methods on most of datasets (excepting the Wine and Iris datasets) at the last learning round. Specifically, the proposed method performed better than the remaining three methods over the majority of the active learning period for the Twonorm, Hayes-Roth, and Letter datasets. Moreover, the D-AL-ELM yielded the fastest learning rate over the first few iterations of the learning process for most datasets. This phenomenon indicates that the new method begins by effectively identifying the most informative and representative samples, unlike the other algorithms. Additionally, the ELM-Entropy approach generally yielded lower accuracy in multiclass classification, failing to surpass the PL-ELM on the Wine, Hayes-Roth, Iris, and Letter datasets. Another interesting observation was that the performance tended to degrade at a certain interval on the Segment dataset. It was considered that the Segment dataset may have a more irregular data structure, confounding the BvSB strategy and deteriorating the result. In cases such as this, a more adaptive stop criterion should be designed to stop the learning program at a more appropriate right time, before output degrades.

Table 2: Details of the optimal parameter settings for the different datasets using four methods.

Dataset	D-AL-ELM		AL-ELM	ELM-Entropy	PL-ELM
	M	λ	M	M	M
Liver	110	0.1	110	110	110
Diabetes	110	0.3	110	110	110
Wdbc	200	0.1	200	200	200
Twonorm	120	0.1	120	120	120
Hayes-Roth	100	0.3	100	100	100
Iris	170	0.9	170	170	170
Wine	60	0.7	60	60	60
Segment	200	0.6	200	200	200
Letter	700	0.4	700	700	700

Table 3 presents the mean classification accuracies of the four methods across the 9 datasets during the learning process. The ALC values for the four methods are further compared in Table 4. The results shown in Tables 3 and 4 indicate that the D-AL-ELM method yielded the best performance among all datasets for the tested methods. As in [13], the ALC metric not only was related to the learning velocity but also had close relationship to the quality of the learning model. The proposed D-AL-ELM outperformed the other methods in terms of ALC, with the AL-ELM performing second best, with an accuracy close to that of the D-AL-ELM on the Wdbc and Segment datasets. For the Wdbc dataset, although the proposed method had a slightly higher ALC value than the AL-ELM, both algorithms yielded the same mean accuracy for the overall learning process.

Finally, Table 5 reports the average time for the learning stage of each algorithm across all datasets. As expected, the PL-ELM was the fastest method because it lacked any criteria for the evaluation of samples. The proposed D-AL-ELM required slightly more learning time than AL-ELM and ELM-Entropy, since it computed both informativeness and representativeness of each instance. Considering the improvement of classification performance, this extra time may be deemed an acceptable tradeoff.

4.1.4. Analysis of Effect of Different Batch Size Values. In this experiment, the performance of the proposed active learning method was further evaluated using different batch sizes (i.e., h and m values).

The new method was tested with different querying sizes by varying the values of h and m, respectively. The remaining experimental settings were the same as in earlier experiments and testing was conducted on two benchmark datasets: Iris and Wine. The M and λ parameters were set as $M = 100$, $\lambda = 0.5$ to observe the performance with different batch sizes. Results are reported in Figures 2 and 3. In Figure 2, m was fixed at 5% of the total number of instances in the original unlabeled set and h was chosen from a candidate set $\{h = 1.1m, 1.2m, 1.5m, 2m, 4m, 5m\}$. In Figure 3, h was fixed at the value of $2m$ and m was chosen from $\{1\%, 2\%, 4\%, 6\%, 8\%\}$. It can be seen from Figure 2 that learning rates at the start of the curve increased with higher h values. Performance on Iris was less sensitive to the h value when enough instances were queried, and learning curves tended to be similar when query numbers and h values were large. In contrast, performance on the Wine dataset was more sensitive to h. This may be a result of the Wine dataset having a more complex distribution, which is difficult to capture. Although the D-AL-ELM performed differently on the two datasets, relatively larger h values were consistently able to obtain favorable performance. On the other hand, this increase in h value leads to a greater computational burden. Figure 3 shows the effects of different values of m on the Iris and Wine datasets. From this, it was observed that convergence can be more easily achieved with small m values. Alternatively, when m is large, more instances can be learned at each iteration and the number of total iterations greatly reduced, although this boost in performance does not provide substantially increased accuracy. In conclusion, optimizing the h and m values is not crucial for the D-AL-ELM, as most values yield similar results. It should be noted, however, that larger h and m are generally recommended.

4.2. Experiment on Multiclass MI EEG Data

4.2.1. Description of EEG Datasets. This section further evaluates the performance of the proposed D-AL-ELM method on multiclass MI EEG data from the BCI Competition IV Dataset 2a [32]. This dataset consists of the EEG signals from 9 subjects who performed 4 tasks, including left hand, right hand, foot, and tongue MI. EEG signals were recorded using 22 electrodes. Each subject underwent a training and testing session, each consisting of 288 trials (a total of 576 trials across the two sessions).

4.2.2. Experimental Setup and Parameter Settings. Data preprocessing was first performed on the raw EEG data. For each trial, features were extracted from the time segment lasting from 0.5 s to 2.5 s after the cue instructing the subject to perform MI. Each trial was first band-pass filtered from 8–30 Hz using a fifth-order Butterworth filter. Next, the dimension of the EEG signal was reduced to a 24-dimension feature set using the one-versus-rest common spatial pattern (OVR-CSP) algorithm [33], which is an effective and popular feature extraction method for EEG multiclassification that computes the features that discriminate each class from the remaining classes. Finally, the features

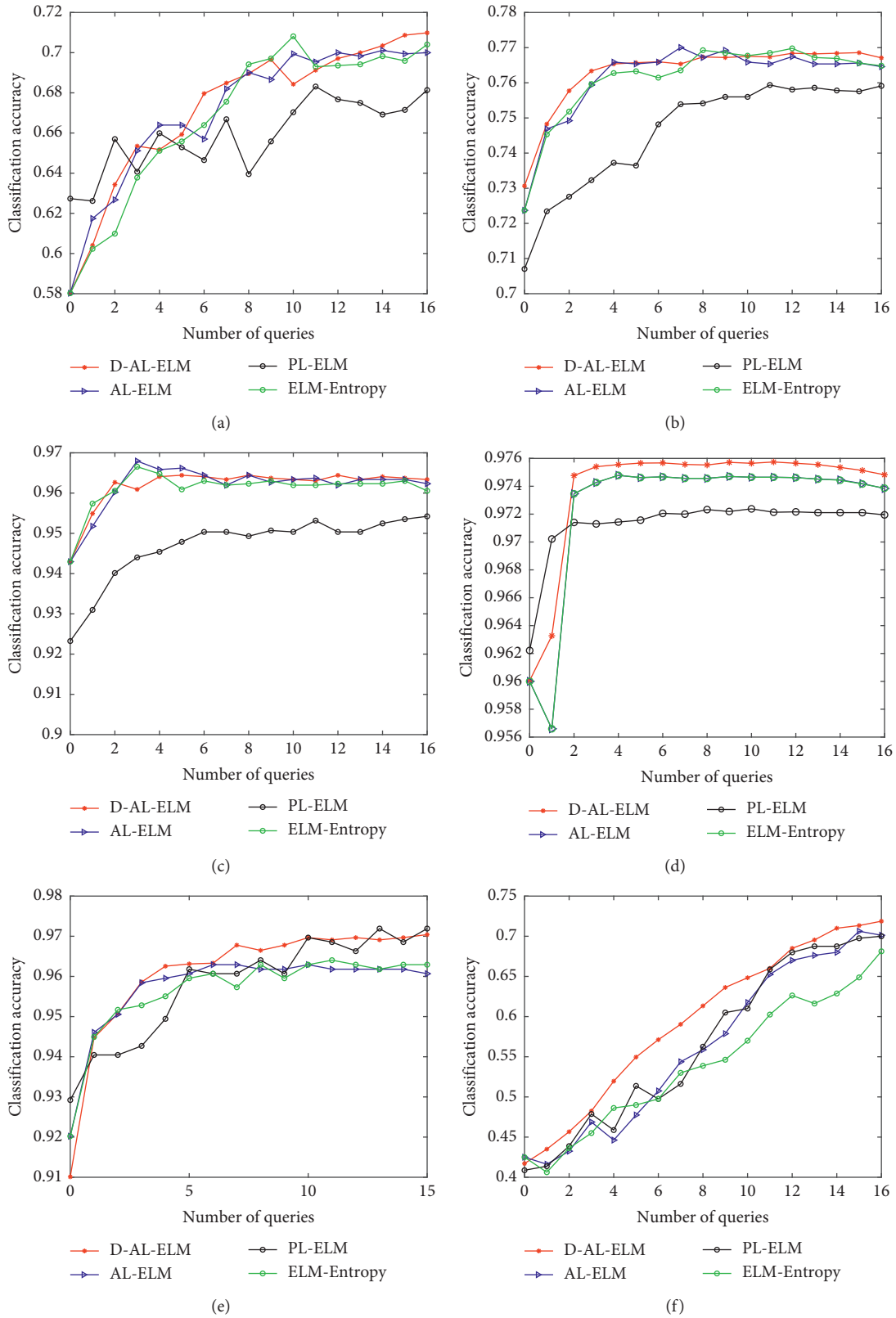

Figure 1: Continued.

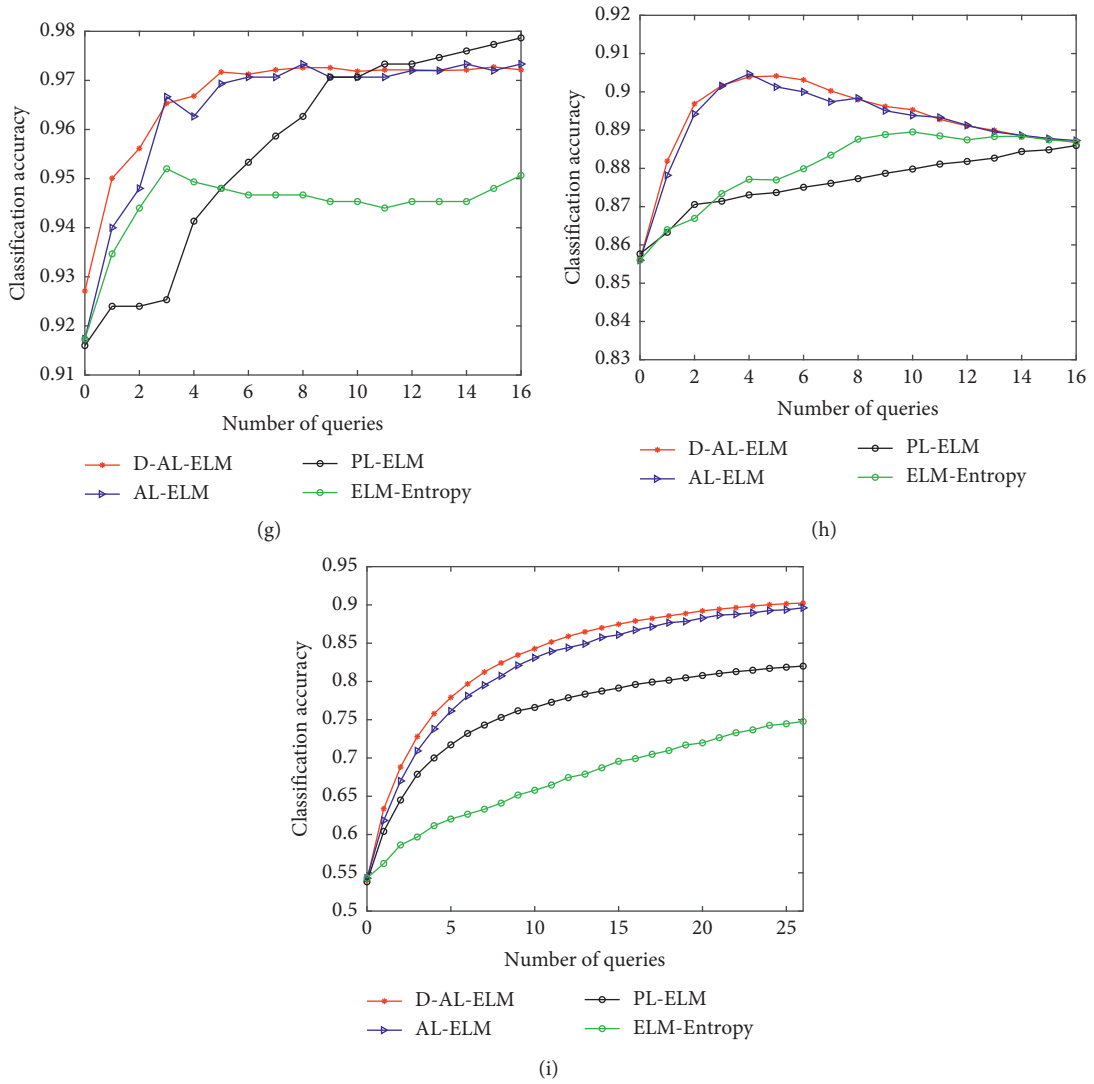

Figure 1: The learning curves of the four different learning algorithms on 9 benchmark datasets. (a) Liver. (b) Diabetes. (c) Wdbc. (d) Twonorm. (e) Wine. (f) Hayes-Roth. (g) Iris. (h) Segment. (i) Letter.

Table 3: Mean accuracy results of the learning processes on 9 datasets (%).

Dataset	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
Liver	**67.59**	67.46	67.14	66.14
Diabetes	**76.31**	76.13	76.12	74.60
Wdbc	**96.18**	**96.18**	96.11	94.69
Twonorm	**97.38**	97.25	97.25	97.13
Wine	**96.08**	95.72	95.64	95.79
Hayes-Roth	**59.43**	56.23	54.03	56.56
Iris	**96.65**	96.43	94.44	95.58
Segment	**89.26**	89.17	88.06	87.63
Letter	**82.89**	81.67	67.09	75.76

Table 4: ALC comparisons of four methods on 9 datasets.

Dataset	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
Liver	**0.7624**	0.7610	0.7572	0.7447
Diabetes	**0.7640**	0.7624	0.7622	0.7469
Wdbc	**0.9624**	0.9623	0.9616	0.9474
Twonorm	**0.9742**	0.9729	0.9729	0.9715
Wine	**0.9622**	0.9584	0.9573	0.9584
Hayes-Roth	**0.5959**	0.5622	0.5395	0.5663
Iris	**0.9676**	0.9655	0.9450	0.9563
Segment	**0.8939**	0.8929	0.8812	0.8766
Letter	**0.8330**	0.8204	0.6718	0.7606

extracted by OVR-CSP were discriminated using the different classification methods.

Optimal selection of the M, λ, and c parameters was performed in the same manner described in Section 4.1.2. The number of hidden nodes M was searched within $\{10, 20, \ldots, 150\}$. For each subject, the first 400 trials were considered as the training set, while the remaining 176 trials were used as the independent testing set [11]. The values for m and h were set at $m = 10$ and $h = 5m$. Finally, experiments included ten runs for each learning method from which average results were calculated.

Table 5: Average running time (s) for each learning algorithm.

Dataset	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
Liver	0.9141	0.7531	0.7719	0.7453
Diabetes	1.2031	1.0438	1.0060	0.9719
Wdbc	1.3484	1.2500	1.2828	1.2234
Twonorm	7.9813	5.3047	5.5875	4.8844
Wine	0.5391	0.4625	0.4625	0.4391
Hayes-Roth	0.5109	0.4516	0.4594	0.4203
Iris	0.5250	0.4469	0.4856	0.4313
Segment	3.6578	3.3906	3312	3.3250
Letter	121.8047	115.5203	123.0641	111.4641

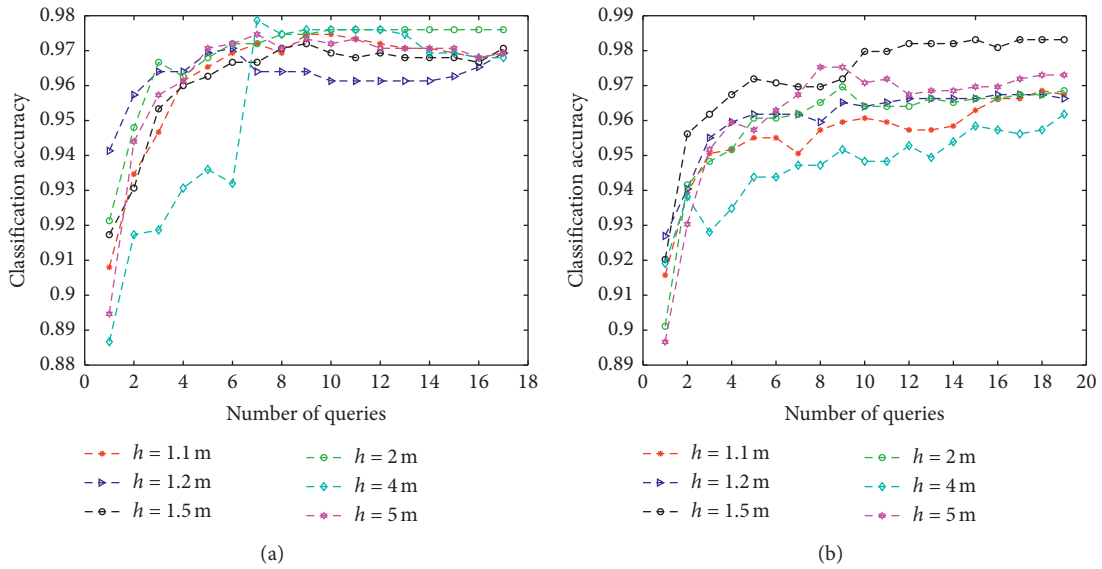

Figure 2: The learning curves of the proposed algorithm with different h values on Iris and Wine. (a) Iris. (b) Wine.

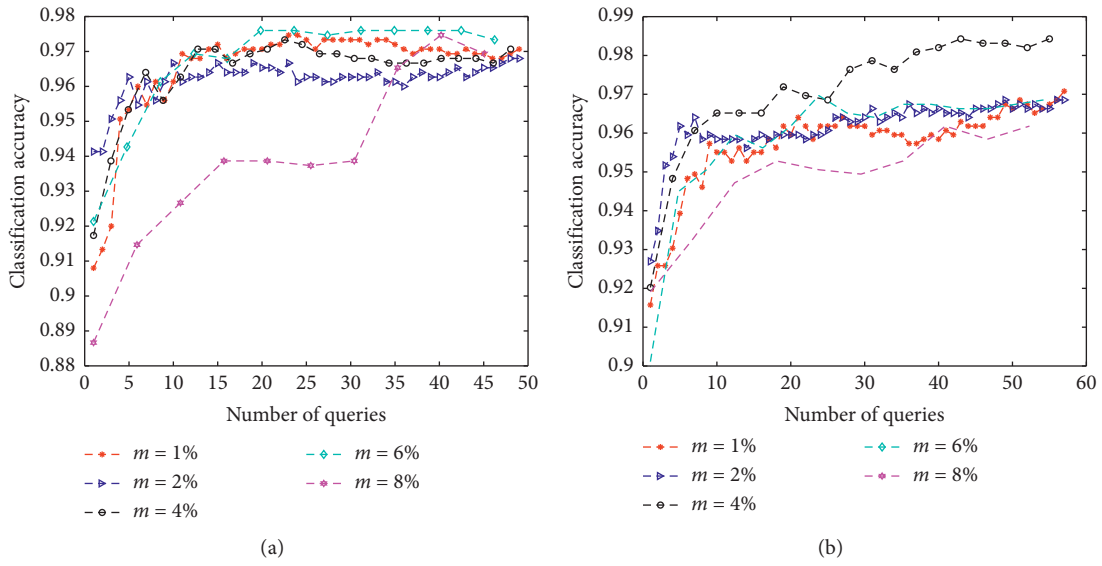

Figure 3: The learning curves of the proposed algorithm with different m values on Iris and Wine. (a) Iris. (b) Wine.

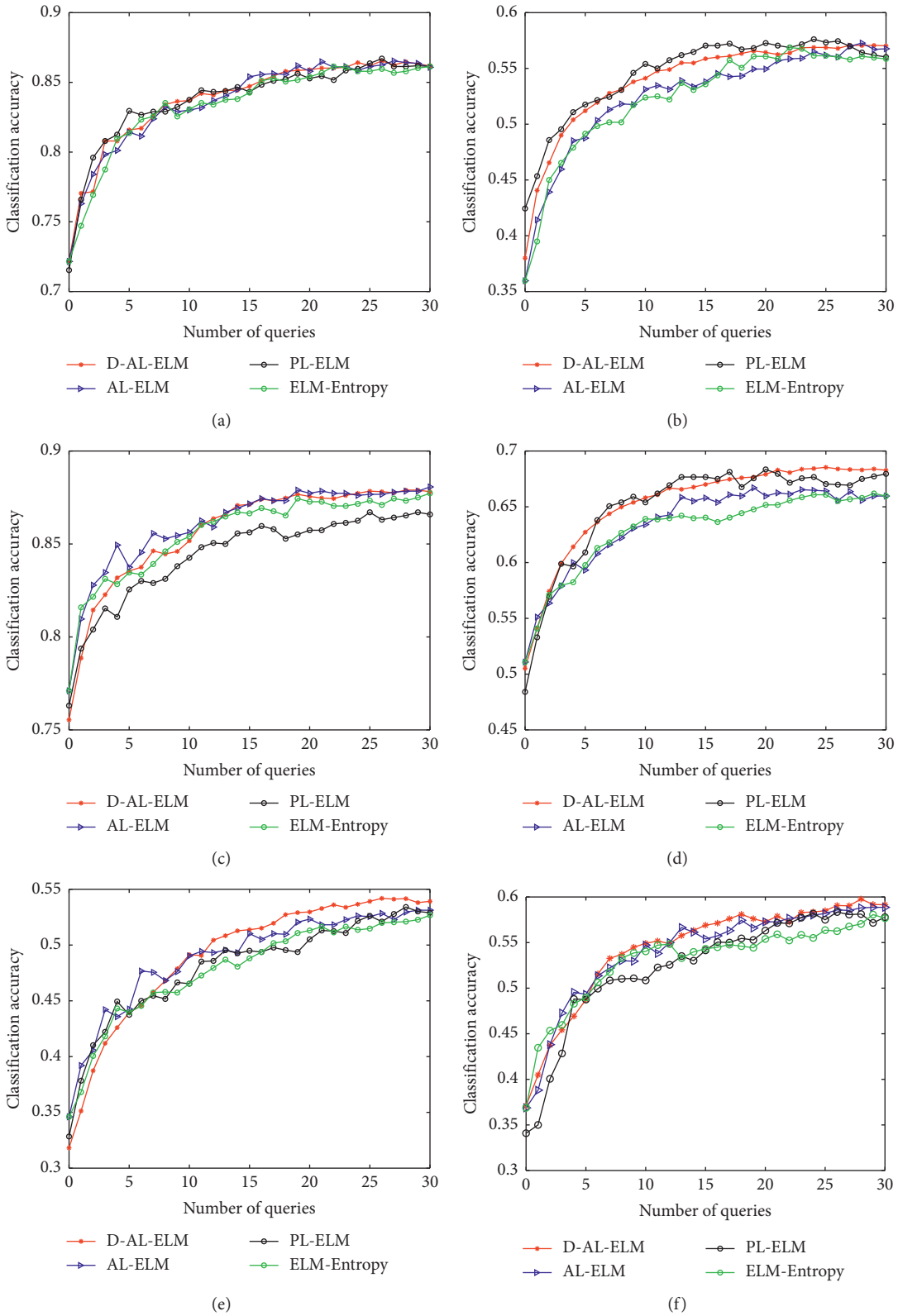

Figure 4: Continued.

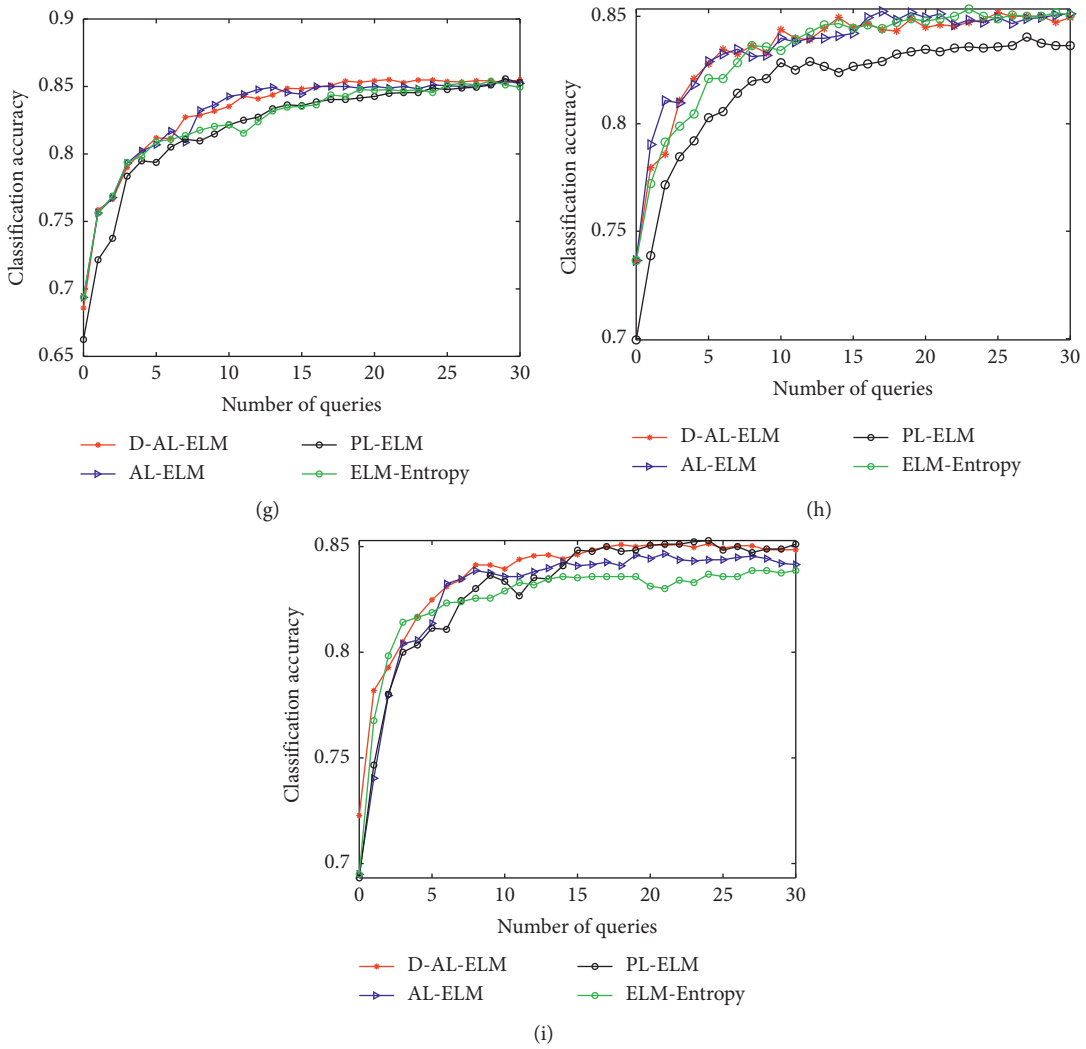

FIGURE 4: Learning curves of the four different learning algorithms on BCI Competition IV Dataset 2a. (a) S1. (b) S2. (c) S3. (d) S4. (e) S5. (f) S6. (g) S7. (h) S8. (i) S9.

4.2.3. Comparisons with Related Algorithms. Figure 4 illustrates the trend lines of classification accuracy when methods were applied to different testing datasets, while Table 6 lists the mean classification accuracies of the four methods during the learning process. Table 7 then provides the ALC results, while Table 8 shows the average running time (s) for the learning stage.

The results show that the performance of D-AL-ELM method is comparable to that of the AL-ELM and better than that of the ELM-Entropy and PL-ELM algorithms for all subjects (except for subject 2 in PL-ELM). Specifically, the proposed method surpassed the AL-ELM approach in 6 of the 9 subjects (1, 2, 4, 5, 6, 9) in terms of the ALC metric. For all 9 subjects, the D-AL-ELM method yielded a mean accuracy of 71.36%, higher than that of AL-ELM (70.92%), ELM-Entropy (70.34%), and PL-ELM (70.51%). These results demonstrate the effectiveness of the D-AL-ELM in selecting both informative and representative instances from unlabeled EEG samples. Additionally, they reveal that the proposed method can calibrate an effective classifier for MI EEG signals without the need for a large number of labeled training samples.

For comparative purposes, Table 8 also provides the average running time of each learning algorithm. Although the D-AL-ELM exhibited slightly longer training time than the other three methods, this may be considered a worthwhile tradeoff for the improved classification performance of the D-AL-ELM.

4.3. Discussion. In these experiments, the proposed D-AL-ELM method exhibited excellent performance in both classification accuracy and computational efficiency, as demonstrated on several benchmark datasets and an experimental MI EEG dataset. When compared to a passive learning-based ELM, D-AL-ELM achieved improved performance by effectively extracting the most valuable unlabeled samples. The D-AL-ELM also outperformed the AL-ELM and ELM-Entropy algorithms, which both employed a single-query strategy. Improvement was seen on all nine

Table 6: Mean accuracy (%) of the learning process on BCI Competition IV Dataset 2a.

Datasets	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
S1	**83.78**	83.64	83.32	83.76
S2	53.91	52.50	52.33	**54.55**
S3	85.66	**86.06**	85.57	84.39
S4	**65.32**	63.48	62.94	64.95
S5	**49.06**	48.90	47.79	48.00
S6	**54.31**	53.96	52.97	52.36
S7	83.15	**83.34**	82.45	81.96
S8	83.46	**83.56**	83.34	81.71
S9	**83.57**	82.80	82.39	82.91
Mean	**71.36**	70.92	70.34	70.51

Table 7: ALC values of the four methods on BCI Competition IV Dataset 2a.

Datasets	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
S1	**81.22**	0.8109	0.8076	81.21
S2	0.5238	0.5100	0.5085	**0.5296**
S3	0.8302	**0.8340**	0.8291	0.8177
S4	**0.6340**	0.6159	0.6105	0.6308
S5	**0.4764**	0.4749	0.4638	0.4662
S6	**0.5276**	0.5241	0.5144	0.5088
S7	0.8066	**0.8085**	0.7996	0.7952
S8	0.8090	**0.8100**	0.8079	0.7923
S9	**0.8103**	0.8032	0.7992	0.8042

Table 8: Average running time (s) of each learning algorithm.

Datasets	D-AL-ELM	AL-ELM	ELM-Entropy	PL-ELM
S1	1.7266	1.4625	1.4594	1.4172
S2	2.5125	2.2813	2.3859	2.2469
S3	1.7219	1.4859	1.4578	1.4234
S4	2.0719	1.7891	1.8328	1.8125
S5	2.0781	1.8281	1.8141	1.7719
S6	1.7609	1.4594	1.4094	1.3609
S7	2.4188	2.1969	2.1734	2.1641
S8	1.5562	1.3375	1.3609	1.3078
S9	1.2906	0.9484	0.9563	0.8906
Mean	1.9042	1.6432	1.6500	1.5995

datasets in Section 4.1, evidencing the ability of the D-AL-ELM to boost overall learning performance by combining the uncertainty and diversity strategies when updating the classifier with the selected samples. In terms of computational efficiency, the slight increase in training time for the D-AL-ELM, as compared to the PL-ELM, AL-ELM, and ELM-Entropy, was negligible in practice, especially when considering the improved classification accuracy. The experimental results then demonstrate that the proposed algorithm can effectively and comprehensively measure the representativeness of samples. Simultaneously, the proposed approach also measures how informative individual examples are, contributing to the improved classifier performance. Combining these factors, suitable instances can be selected for classifier construction.

Finally, the effectiveness of the D-AL-ELM was shown in its application to an experimental multiclass MI task from the BCI Competition IV Dataset 2a. Due to the low signal-to-noise ratio of EEG data, the applied algorithms struggled to generate adequate results. Consequently, hand-designed features were first extracted from the raw EEG data using the OVR-CSP approach, and the different AL algorithms were then used to further extract the unlabeled samples and calibrate a robust classifier. For subjects S1, S3, S7, S8, and S9, the D-AL-ELM yielded an acceptably high mean classification accuracy of over 80% for the whole learning process. Unfortunately, all tested methods performed poorly on subject S5. The proposed algorithm was only able to achieve 49.06% accuracy which, though insufficient, still ranked the highest among the applied methods.

5. Conclusion

In this paper, a novel active learning method with ELM, the D-AL-ELM, was developed for multiclassification. This new algorithm combines the uncertainty and diversity strategies and effectively reduces the expense and time cost of obtaining labeled data manually. For each sample, the proposed algorithm employs a BvSB strategy to measure informativeness and the cosine angle distance to measure diversity. The modified diversity measure not only estimates the diversity between the limited labeled training data and previously selected unlabeled samples, but also calculates the similarity among previously selected samples. Experimental results from several benchmark datasets and the multiclass MI EEG data from BCI Competition IV Dataset 2a were then used to verify the efficacy of the proposed D-AL-ELM algorithm. These results indicate that the performance of the proposed algorithm is consistently better than, or at least comparable to, that of other popular active learning techniques. Future work will then aim to develop online learning for the D-AL-ELM [23, 34]. In addition, an adaptive stopping criterion may be applied to promote the efficiency of the D-AL-ELM and improve its abilities for the classification and evaluation of MI EEG signals.

Data Availability

The BCI Competition IV Dataset 2a was used in our study, which is publicly available via the following link: http://www.bbci.de/competition/iv/.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61871427 and 61671197). The authors would like to acknowledge the BCI Competition IV Dataset 2a which was used to test the algorithms proposed in this study.

References

[1] F. Lotte, L. Bougrain, A. Cichocki et al., "A review of classification algorithms for EEG-based brain-computer interfaces: a 10-year update," *Journal of Neural Engineering*, vol. 15, no. 3, p. 031005, 2018.

[2] P. Gonzalez-Navarro, M. Moghadamfalahi, M. Akcakaya, and D. Erdogmus, "Spatio-temporal EEG models for brain interfaces," *Signal Processing*, vol. 131, pp. 333–343, 2017.

[3] K. K. Ang and C. Guan, "EEG-based strategies to detect motor imagery for control and rehabilitation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 4, pp. 392–401, 2017.

[4] R. Zhang, Y. Li, Y. Yan et al., "Control of a wheelchair in an indoor environment based on a brain-computer interface and automated navigation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 1, pp. 128–139, 2016.

[5] Q. She, H. Gan, Y. Ma et al., "Scale-Dependent signal identification in low-dimensional subspace: motor imagery task classification," *Neural Plasticity*, vol. 2016, p. 15, 2016.

[6] R. Li, T. Potter, W. Huang et al., "Enhancing performance of a hybrid EEG-fNIRS system using channel selection and early temporal features," *Frontiers in Human Neuroscience*, vol. 11, p. 462, 2017.

[7] P. Gaur, R. B. Pachori, H. Wang, and G. Prasad, "A multi-class EEG-based BCI classification using multivariate empirical mode decomposition based filtering and riemannian geometry," *Expert Systems with Applications*, vol. 95, no. 1, pp. 201–211, 2018.

[8] J. Li and L. Zhang, "Active training paradigm for motor imagery BCI," *Experimental Brain Research*, vol. 219, no. 2, pp. 245–254, 2012.

[9] Y. Freund, H. S. Seung, E. Shamir, and N. Tishby, "Selective sampling using the query by committee algorithm," *Machine Learning*, vol. 28, no. 2/3, pp. 133–168, 1997.

[10] S. Kee, E. Del Castillo, and G. Runger, "Query-by-committee improvement with diversity and density in batch active learning," *Information Sciences*, vol. 454-455, pp. 401–418, 2018.

[11] V. Lawhern, D. Slayback, D. Wu et al., "Efficient labeling of EEG signal artifacts using active learning," in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3217–3222, Kowloon, China, October 2015.

[12] M. Chen, X. Tan, J. Q. Gan et al., "A batch-mode active learning method based on the nearest average-class distance (NACD) for multiclass brain-computer interfaces," *Journal of Fiber Bioengineering & Informatics*, vol. 7, no. 4, pp. 627–636, 2014.

[13] H. Yu, C. Sun, W. Yang, X. Yang, and X. Zuo, "AL-ELM: one uncertainty-based active learning algorithm using extreme learning machine," *Neurocomputing*, vol. 166, pp. 140–150, 2015.

[14] R. Wang, S. Kwong, Q. Jiang et al., "Active learning based on single-hidden layer feed-forward neural network," in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, pp. 2158–2163, Kowloon, China, October 2015.

[15] Z. Qiu, D. J. Miller, and G. Kesidis, "A maximum entropy framework for semisupervised and active learning with unknown and label-scarce classes," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 917–933, 2017.

[16] R. Chattopadhyay, Z. Wang, W. Fan, I. Davidson, S. Panchanathan, and J. Ye, "Batch mode active sampling based on marginal probability distribution matching," *ACM Transactions on Knowledge Discovery from Data*, vol. 7, no. 3, pp. 1–25, 2013.

[17] H. Ibrahim, K. Abbas, H. Imali et al., "Multiclass informative instance transfer learning framework for motor imagery-based brain-computer interface," *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 6323414, 12 pages, 2018.

[18] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of Machine Learning Research*, vol. 2, no. Nov, pp. 45–66, 2001.

[19] S. C. H. Hoi, R. Jin, J. Zhu, and M. R. Lyu, "Semisupervised SVM batch mode active learning with applications to image retrieval," *ACM Transactions on Information Systems*, vol. 27, no. 3, pp. 1–29, 2009.

[20] Q. Zhang and S. Sun, "Multiple-view multiple-learner active learning," *Pattern Recognition*, vol. 43, no. 9, pp. 3113–3119, 2010.

[21] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006.

[22] G. Huang, G.-B. Huang, S. Song, and K. You, "Trends in extreme learning machines: a review," *Neural Networks*, vol. 61, pp. 32–48, 2015.

[23] Y. Zhang and M. J. Er, "Sequential active learning using meta-cognitive extreme learning machine," *Neurocomputing*, vol. 173, no. P3, pp. 835–844, 2016.

[24] B. Du, Z. Wang, L. Zhang et al., "Exploring representativeness and informativeness for active learning," *IEEE Transactions on Cybernetics*, vol. 47, no. 1, pp. 14–26, 2017.

[25] Y. Gu, S. C. Chiu, and Z. Jin, "Active learning combining uncertainty and diversity for multi-class image classification," *IET Computer Vision*, vol. 9, no. 3, pp. 400–407, 2015.

[26] S.-J. Huang, R. Jin, and Z.-H. Zhou, "Active learning by querying informative and representative examples," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 1936–1949, 2014.

[27] G. B. Huang, H. Zhou, X. Ding et al., "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems Man & Cybernetics Part B*, vol. 42, no. 2, pp. 513–529, 2012.

[28] A. J. Joshi, F. Porikli, and N. P. Papanikolopoulos, "Scalable active learning for multiclass image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2259–2273, 2012.

[29] J. Alcalá-Fdez, A. Fernández, J. Luengo et al., "KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework," *Journal of Multiple-Valued Logic & Soft Computing*, vol. 17, pp. 255–287, 2011.

[30] D. Dua and E. Karra Taniskidou, *UCI Machine Learning Repository*, http://archive.ics.uci.edu/ml/, 2017.

[31] J. Cao, K. Zhang, M. Luo, C. Yin, and X. Lai, "Extreme learning machine and adaptive sparse representation for image classification," *Neural Networks*, vol. 81, pp. 91–102, 2016.

[32] M. Tangermann, K. R. Müller, A. Aertsen et al., "Review of the BCI competition IV," *Frontiers in Neuroscience*, vol. 6, p. 55, 2012.

[33] M. Meng, J. Zhu, Q. She et al., "Two-level feature extraction method for multi-class motor imagery EEG," *Acta Automatica Sinica*, vol. 42, no. 12, pp. 1915–1922, 2016.

[34] J. S. Lim, S. Lee, and H. S. Pang, "Low complexity adaptive forgetting factor for online sequential extreme learning machine (OS-ELM) for application to nonstationary system estimations," *Neural Computing & Applications*, vol. 22, no. 3-4, pp. 569–576, 2013.