Pattern Recognition 150 (2024) 110341

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

A patch distribution-based active learning method for multiple instance Alzheimer's disease diagnosis

Tianxiang Wang, Qun Dai *

College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

ARTICLE INFO

Keywords:
Multi-instance learning
Active learning
Alzheimer's disease
Attention mechanism

ABSTRACT

Medical data, particularly the complex brain imaging structures, acquisition presents significant difficulties and high diagnostic expenses, resulting in a scarcity of the trainable samples in the real-world scenarios. To overcome this limitation, we present an active learning-based sampling strategy that selects the most informative samples from the unlabeled candidate sample pool for expert annotation, leading to high classification performance with a reduced number of training samples. This study adopts a patch-level perspective and introduces a multi-instance learning framework for Alzheimer's Disease diagnosis. Initially, a patch pre-selection module is designed to identify pathology-prone regions while excluding background areas and irrelevant information. Subsequently, an inner-patch local attention mechanism block and an outer-patch global attention mechanism block are developed to enhance the extraction of discriminative local and global information by the network model. Finally, an active learning sampling strategy is devised to minimize the costs associated with data acquisition and expert annotation in medical domain. The effectiveness of the proposed network framework and active learning strategy was validated through four sets of control experiments on the ADNI dataset.

1. Introduction

Alzheimer's disease (AD) is a neurodegenerative disease with an insidious onset and progressive development [1]. Although Alzheimer's disease itself is not a fatal illness, the disease is prone to complications such as pneumonia, bed sores, and malnutrition [2]. Studies have shown that the average lifespan of Alzheimer's patients is 5-10 years shorter than that of other healthy peers [1]. Therefore, the harm of this disease should not be underestimated. Up to now, there is no cure for Alzheimer's disease, and the progression of the disease can only be slowed down through improving the patient's quality of life, medication, and non-pharmacological treatments [2]. Early prevention and treatment remain the optimal strategy for addressing this symptom [3]. The entire process of Alzheimer's disease can be roughly classified into four diagnostic categories: normal cognition (NC), stable mild cognitive impairment (sMCI), progressive mild cognitive impairment (pMCI), and Alzheimer's disease (AD) [4]. It is beneficial to investigate a model that can assist neurology experts in accurately and efficiently detecting early signs of Alzheimer's disease.

Due to its non-ionizing radiation, absence of bone artifacts, multi-faceted and multi-parametric imaging capabilities, high soft tissue resolution, and the ability to display vascular structures without the need for contrast agents, Magnetic Resonance Imaging (MRI) has significant advantages [5]. Therefore, most existing brain scans of Alzheimer's patients are stored in the form of structured MRI (sMRI) [6]. The traditional sMRI-based diagnostic approaches for Alzheimer's disease typically involve segmenting the entire MR image into multiple regions of varying scales to facilitate the extraction of local abnormal features [7]. Currently, the sMRI-based researches can be broadly divided into four classes: (1) Voxel-level [8], (2) Region-level [9], (3) Sliced-level [10], and (4) Patch-level [6]. Patient images saved in this way can be seen as the domain of multi-instance learning, i.e., in the scan images of Alzheimer's disease patients, only local information contains pathological features of Alzheimer's disease, while the rest of the imaging are indistinguishable from those of ordinary patients. In this study, we model the problem of Alzheimer's disease detection from a patch-level perspective. After that, based on a multi-instance learning framework, we introduce an in-patch local attention mechanism and an out-of-patch global attention mechanism to construct the network. This strategy not only reduces the interference of invalid information in the imaging, but also improves the model's focus on the more important instances in the 'bag'.

* Corresponding author.
E-mail address: daiqun@nuaa.edu.cn (Q. Dai).

https://doi.org/10.1016/j.patcog.2024.110341
Received 14 July 2023; Received in revised form 8 February 2024; Accepted 12 February 2024
Available online 14 February 2024

With the advancement of datamining technology and scientific level, countless deep learning-based methods have been applied to real-world scenarios with outstanding performance [11]. The deep learning-based models can fit arbitrary mapping from a large number of trainable data [12]. However, the condition of large amount of trainable data is not available in medical scenarios due to the high cost of scanning medical images and the difficulty of manual annotation [13]. To address this issue, strategies such as weakly supervised learning [14], semi-supervised learning [15], transfer learning [16], data augmentation [17], ensemble learning [11], active learning [18] have emerged. Among them, weakly supervised learning uses labels that are less accurate or less complete than fully labeled data to train the model, which can effectively reduce the cost and time of annotation [14]. However, in this approach, weak labels may not provide enough information to train an accurate model, resulting in performance degradation, additionally, the noise in weak labels can also have a significant negative impact on the training and prediction results of the model [19]. Semi-supervised learning is a technique that utilizes a limited amount of labeled data and abundant unlabeled data for training and prediction, which can directly reduce the cost of annotation [15]. Nevertheless, this approach demands a large quantity of high-quality unlabeled data, whose acquisition in medical scenarios may be challenging or even infeasible. Therefore, the semi-supervised learning strategy may not always represent the optimal solution for medical applications [20]. Transfer learning leverages pre-labeled datasets and models to quickly train a high-performance model in a new domain, thereby reducing the annotation cost of new datasets [16]. Still, in transfer learning, if appropriate source and target domains are not identified, the accuracy of the model may be significantly compromised [21]. Data augmentation is a methodology that expands and transforms existing labeled data to increase sample diversity, reduce annotation costs, and escalate the number of trainable samples [17]. Yet, the augmented data may alter the sample distribution in the training set, leading to a decrease in accuracy on the test set [22]. Ensemble learning combines the predictions of multiple models to improve their accuracy and robustness, resulting in a reduced impact of mislabeled data [11]. However, ensemble learning is computationally expensive, and its final prediction results depend on the diversity among the models being integrated. Active learning reduces annotation costs by selecting the most informative samples for human experts to annotate, which can effectively address or avoid the aforementioned issues [18]. This strategy is particularly suitable for medical scenarios with a high annotation difficulty and a limited number of trainable samples.

Accordingly, our work designs an instance-based patch-level distributional active learning strategy for the diagnosis of Alzheimer's disease. A summary of our main contributions can be outlined as follows:

(1) Due to the presence of a large amount of background and invalid areas in the entire MR image, inputting the whole image into the network would lead to increased time and space consumption and introduce substantial noise. Therefore, this study treats each patch as a feature and proposes a patch pre-selection module based on the Relief model. However, traditional Euclidean distance used to measure the difference between patches may cause the loss of spatial information within the patch. To overcome this issue, the study introduces a block-wise Hash difference measure to replace Euclidean distance, which significantly preserves spatial and structural information within each patch. Subsequently, the importance of each patch is calculated, and blocks with relatively higher significance are selected as candidate blocks for subsequent experiments in the model.
(2) Following the patch pre-selection module, in order to further reduce the negative impact of irrelevant instances in the 'bag', we formulate the problem of Alzheimer's Disease diagnosis with sMRI as a patch-based multi-instance problem. This paper designs a Patch-Level Global and Local Attention-based Multi-Instance Deep Learning Model (PLGLA) that utilizes attention mechanisms to enhance Alzheimer's disease diagnosis performance. In addition, this model also lays the foundation for learning instance-level data distribution tasks in the subsequent active learning module.
(3) To address the challenges posed by limited medical image samples and difficult labeling, this paper constructs a Patch-Level Instance Distribution-based Active Learning Strategy (PIDAC). Initially, we design a network that can implicitly learn the Gaussian distribution of several blocks with the highest attention in the PLGLA model for different categories of samples. We then leverage the additivity property of Gaussian distribution, to calculate the data distribution of these blocks among various categories. Finally, we select the least discriminative samples from the candidate data set based on both sample-level and decision-level Gaussian distributions, and incorporate them into the training set to minimize labeling costs.

The remainder of this paper is organized as follows: Section II outlines the related works on Multi-Instance Learning, Attention Mechanism, and Active Learning. Section III presents the materials studied in this research and introduces the proposed PLGLA and PIDAC models. The experimental results and analysis of experiments are presented in Section IV. In Section V, further discussions are conducted on the designed model from the perspectives of statistics, model efficiency, visualization, and generalizability, and Section VI provides a summary of this paper.

2. Related works

2.1. Multi-instance learning

In medical scenarios, it is difficult to accurately annotate all affected areas, and there is not much difference between the other areas of patients and healthy individuals, therefore, many disease diagnosis problems in medical images can be considered as a category of multi-instance learning tasks [6]. Specifically, multi-instance learning is a supervised learning method, in which the input data consists of bags of multiple instances instead of single instances. These bags are labeled as positive or negative instead of each instance being labeled [23,24]. The mathematical model can be defined as: Suppose that a training dataset $D = \{B_i, y_i\}$, where B_i denotes a bag consisting of multiple instances, i.e., $B_i = \{x_{i,1}, x_{i,2}, \cdots, x_{i,n}\}$, $y_i \in \{0,1\}$ is the label of B_i, $y_i = 1$ represents that the bag B_i is positive, while $y_i = 0$ indicates that the bag B_i is negative. For each bag, there are n instances in the i th bag, and the number of instances in different bags may vary. For the instance $x_{i,j}$ in the B_i, its corresponding label $y_{i,j}(1 \le j \le n)$ is unknown. When the label of all instances in the B_i are 0, i.e., $\sum y_{i,j} = 0, (y_{i,j} \in \{0,1\}, j = 1,2,\cdots,n)$, then the label y_i w.r.t B_i is 0, i.e., $y_i = 0$, otherwise $y_i = 1$. The mission of multi-instance learning is to achieve a predictor $f : B \rightarrow y$ from D.

In recent years, numerous scholars have explored the application of multi-instance learning, which has shown promising performance in the field of computer-aided medical diagnosis [25]. Liu et al. [26] proposed a landmark-based deep multi-instance learning neural network for brain disease diagnosis, this framework predicted the final disease status of the patient by capturing the local structural information conveyed by the image patches located by landmark localization and the overall structural information generated by all detected landmarks. Couture et al. [27] designed a novel approach, referred to as MIL pooling, this method was proposed utilizing the quantile function to summarize predictions from smaller regions into an image-level classification for breast tumor histology. Zhu et al. [6] first designed a patch selection strategy, then they modeled the patch-based sMRI brain images as a multi-instance learning problem and a dual attention module was designed to further increase the focus on the lesion areas. The

above-mentioned research considers medical disease detection tasks as multi-instance learning tasks to reduce the negative impact of non-lesion areas on the final prediction results. Inspired by these study contents, this paper formulates the problem of Alzheimer's Disease diagnosis with sMRI as a patch-based multi-instance problem and designs an inner-patch local attention module and an outer-patch global attention module to simultaneously focus on multiple lesion areas. However, while MIL approaches have shown promise, the interpretation of their decisions can be challenging due to the lack of explicit localization information [6].

2.2. Attention mechanism

In the field of medicine, the task of medical image processing usually focuses on specific diseases or lesions, so the attention mechanism can help the model better focus on key areas and improve diagnostic accuracy [6], furthermore, thanks to the excellent interpretability of the attention mechanism, this module can assist medical experts in some cases in identifying disease regions that are typically difficult to perceive [28]. So far, numerous attention-based derivative models have emerged based on the theory of generalized attention mechanism and have been successfully applied to various medical scenarios [3]. Cheng et al. [29] proposed a dual-attention domain adaptation segmentation network for cross-modality medical image segmentation, which normalizes the domain adaptation module from both spatial and category perspectives through two attention maps, thereby improving the model's generalization ability and accuracy. The network performs well in cross-modality medical image segmentation tasks. Shang et al. [30] designed a gated multi-attention feedback network for the medical image Super-resolution task. The network used a gate-activated multi-feedback network as the backbone to extract hierarchical features. Additionally, a Hierarchical Attention Feature Extraction module is introduced to refine the feature maps, and a Channel-Spatial Attention Reconstruction module is established to enhance the representation ability of semantic feature maps. Since the high adaptability of attention mechanisms to medical images, this paper presented an inner-patch local attention module and an outer-patch global attention module to further enhance the focus on lesion area.

2.3. Active learning

Although we can consider attempting to combine multiple-instance learning with attention mechanisms to enhance focus on the pathological regions and attenuate noise interference from normal regions, however, there is still a pressing issue at hand. Medical sample collection poses challenges, with high costs for annotation and a limited availability of trainable samples [31]. Therefore, the paramount concern is how to train a classifier using a reduced number of samples while attaining performance that approaches that of a classifier trained on the complete dataset. Fortunately, Active Learning has become a promising approach to address these issues [18]. Active Learning is a semi-supervised learning method that aims to improve the performance of classifiers by minimizing the number of labeled samples [32]. In Active Learning, the classifier can actively select some unlabeled samples for labeling to better learn the classifier. In popular terms, assuming there are n unlabeled samples, each represented by m features, the input of active learning can be represented as an $n \times m$ matrix X, where each row represents an unlabeled sample and each column represents a feature. Meanwhile, the goal of active learning is to learn a classifier $f(X)$ to predict the labels Y of the samples. In active learning, the classifier selects some unlabeled samples to be labeled in order to better learn the classifier. In other words, the classifier selects some samples to be labeled based on some active learning strategies, and then adds these labeled samples to the labeled sample set to retrain the classifier. This process is iterated until the performance of the classifier reaches the expected level or the number of labeled samples reaches the preset upper limit [33].

Therefore, based on the data distribution of the positive and negative datasets, this article selects the most uncertain samples with the greatest difference in distribution from the known positive and negative datasets. In addition, to increase the diversity of the sampled samples, we add a constraint that the sampling can make the distribution of the current labeled dataset (comprehensive consideration of positive or negative classes) change the most. Finally, the designed patch-based active learning strategy considers multiple instances (patches) in a sample, which minimizes the interference of noisy data on sampling.

3. Our method

In this section, we provide the mathematical model and implementation details of our proposed method. The framework of the developed method is illustrated in Fig. 1 and consists of three main components: (1) A patch pre-selection model, (2) A Patch-Level Global and Local Attention-based Multi-Instance Deep Learning network (PLGLA), and (3) A Patch-Level Instance Distribution-based Active Learning strategy (PIDAC).

3.1. The patch pre-selection model

Given that the original image comprises a substantial proportion of background areas, and pathological information is typically confined to small, localized regions, processing the entire image directly not only incurs significant time and space costs, but also undermines the model's capacity to concentrate on critical areas. Therefore, it is necessary to design a patch pre-selection model to reduce the negative impact of unnecessary information. Relief, as a classic feature selection algorithm, aims to remove irrelevant and redundant features. For a binary classification problem, the mathematical model of this algorithm can be represented as follows:

Suppose that a training dataset $D = \{(x_i, y_i)\}_{i=1}^{n}$, where n is the number of training data, x_i denotes the i th sample in the training dataset, which is represented by an m-dimensional feature vector, i.e., $x_i = \{x_i^1, x_i^2, \cdots, x_i^m\}$, and $y_i \in \{0, 1\}$ indicates the label of x_i. Therefore, the set of positive samples and negative samples in D can be expressed as $D_p = \{x_i | x_i \in D, y_i = 1\}$ and $D_n = \{x_i | x_i \in D, y_i = 0\}$, respectively. Then, for a given hyper-parameter k, the weight of each feature x^j $(1 \le j \le m)$ can be computed as follow:

$$w(j) = w(j) - \frac{\mu}{k}\sum_{l=1}^{k} dif\left(x_i, N_l^+(x_i)\right) + \frac{\mu}{k}\sum_{l=1}^{k} dif\left(x_i, N_l^-(x_i)\right), \tag{1}$$

where $N_l^+(x_i) \in D_p$ and $N_l^-(x_i) \in D_n$ denote the l-th nearest distance to the same or different class as x_i, respectively. $\mu = 1$, if $y_i = 1$, and $\mu = -1$, if $y_i = 0$. $dif(x_i, x_j)$ indicates the diversity between x_i and x_j, and in the conventional Relief model, the Euclidean distance was used to calculate the difference between two samples. For the Eq. (1), the feature weight w will be continuously updated with the increase of iteration times. The specific algorithm description and parameter introduction can be found in [34].

The Relief algorithm offers several advantages. Firstly, it is capable of handling high-dimensional datasets and performs well in such datasets. Secondly, the Relief algorithm measures the importance of features based on the distance between samples, effectively filtering out redundant and irrelevant features, thereby improving the model's generalization performance. Thirdly, the Relief algorithm calculates feature weights based on the distance between instances, without involving issues of probability distribution shift. Finally, the Relief algorithm does not rely on prior statistical knowledge or machine learning models. Hence, we propose to split the original MR image into several patches of the same size, and treat these blocks as features. Subsequently, a patch selection algorithm is designed based on the Relief algorithm.

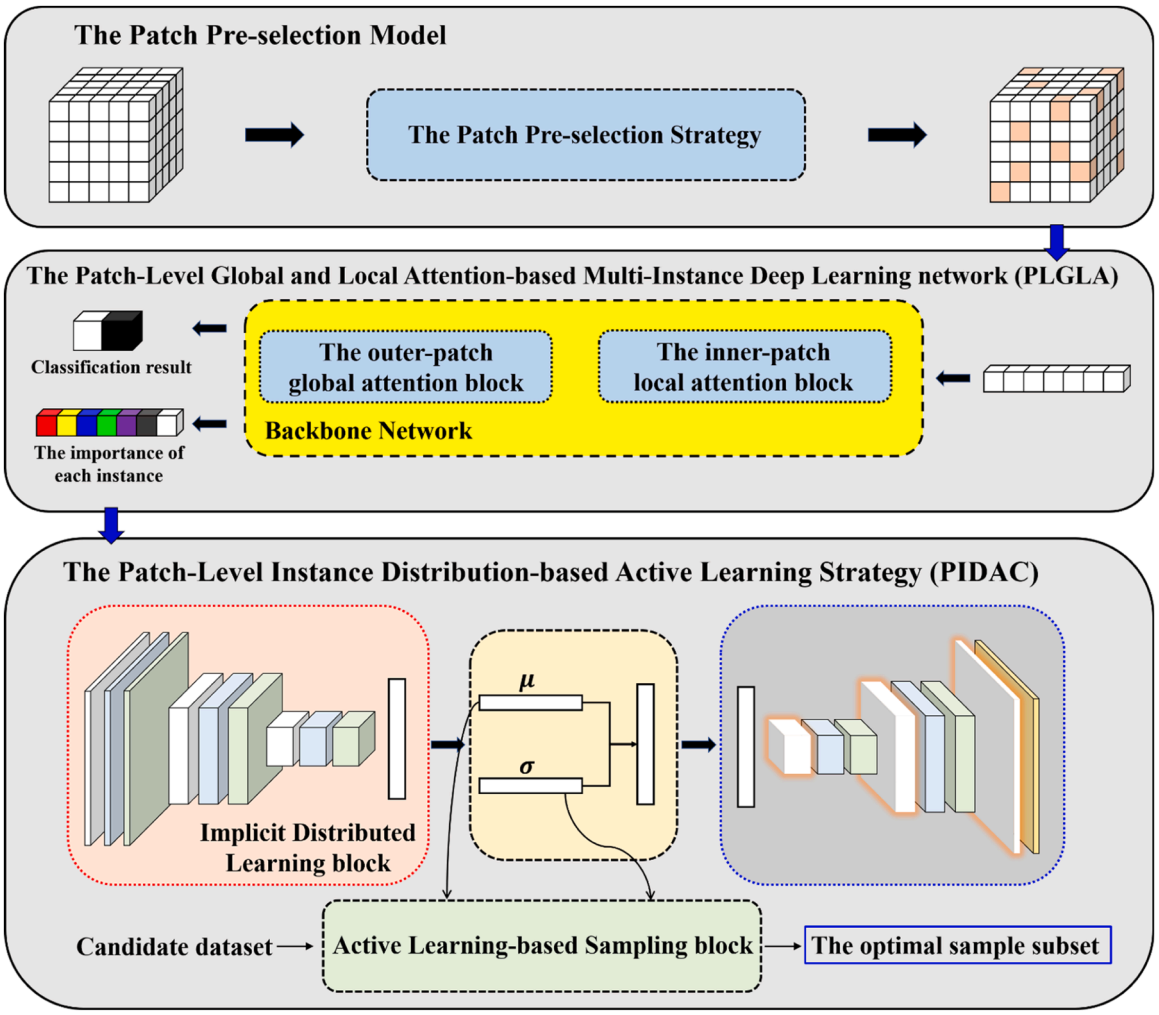


Fig. 1. The framework of the developed method.

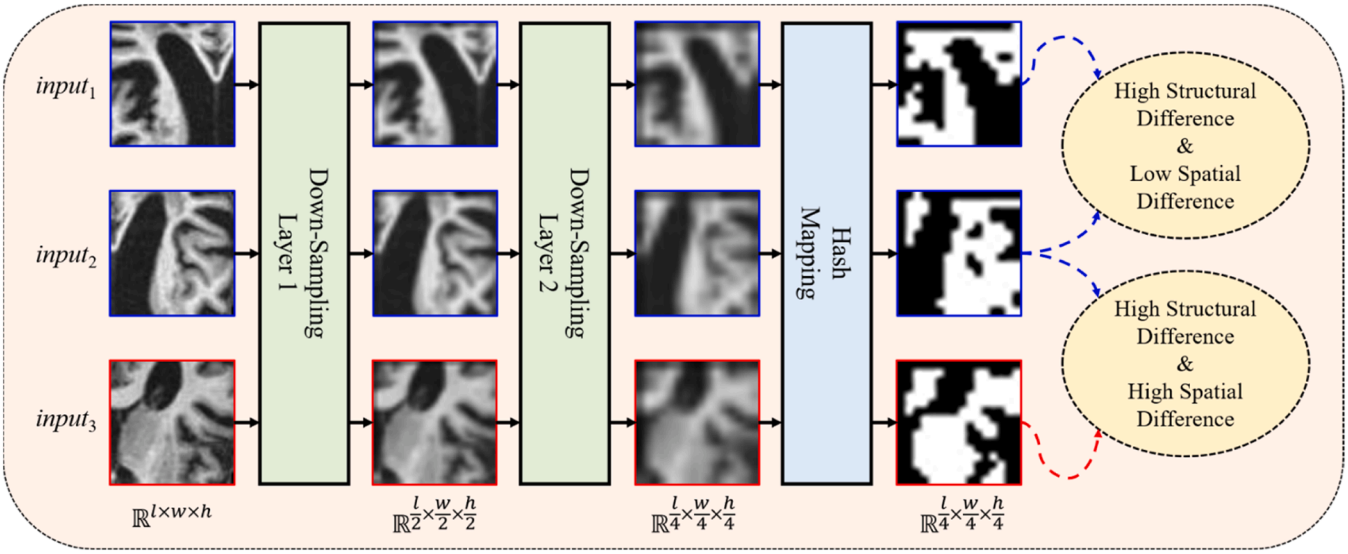


Fig. 2. The computational process of hash-based difference. input1 and input2 are two nearly symmetrical regions of the same AD patient, but the structural differences between these two regions are significant while the spatial differences are minimal. input2 and input3 are images of an AD patient and a healthy patient, respectively, so the structural and spatial differences between these two images are relatively larger.

In our approach, we treat a spatial patch as a feature. However, using the Euclidean distance to measure the difference between two patches may result in significant loss of spatial information. To address this issue, this paper considers introducing Hash difference instead of Euclidean distance. Due to the large size of the original image and the intricate complexity of effective structures and noise information, we employ two consecutive average down-sampling operations to extract more macroscopic spatial information, reduce noise interference, and simultaneously reduce the image size.

For a given MR image $I \in \mathbb{R}^{l\times w\times h}$, l, w, h indicate the length, weight, and height of I, suppose that a down-sampling operation with a stride size of 2 is $\otimes_2$, and a convolution kernel size of (3,3,3) is $\mathscr{K}(3,3,3)$, then we can obtain a preprocess image $I^{'} \in \mathbb{R}^{l/4\times w/4\times h/4}$ after two consecutive average down-sampling operations, which is defined as Eq. (2).

$$I^{'} = I\otimes_2\mathscr{K}(3,3,3)\otimes_2\mathscr{K}(3,3,3). \tag{2}$$

In order to facilitate subsequent difference calculations, we performed a discretization operation on $I^{'}$, which sets the elements corresponding to positions greater than the mean to 1 and those less than or equal to the mean to 0. The specific mathematical model is shown in Eq. (3), and the corresponding detailed example can be found in Fig. 2.

$$I^{''}(i,j,k) = \begin{cases} 1, I^{'}(i,j,k) > mean(I^{'}) \\ 0, I^{'}(i,j,k) \le mean(I^{'}) \end{cases}, 1 \le i \le \frac{l}{4}, 1 \le j \le \frac{w}{4}, 1 \le k \le \frac{h}{4}, \tag{3}$$

where $mean(I^{'})$ indicates the average pixel value of patch $I^{'}$. In comparison, the function described in Eq. (3) shares similarities with binary segmentation algorithms. However, it possesses a simpler principle and operates at a higher speed. It is sufficient for application to brain MR images with fewer details, a single structure, and a significant contrast between foreground and background regions. Consequently, this formulation enables the extraction of cerebral structures from individual patches, thereby generating corresponding hash values for each patch. These hash values can subsequently be employed in the computation of inter-patch similarity.

After that, we calculate the difference between two patches from both spatial difference and structural difference perspectives simultaneously. Suppose that two patches I_a and I_b, then we can get two preprocessed patches $I_a^{''}$ and $I_b^{''}$ from the Eqs. (2) and (3). The spatial difference and structural difference between I_a and I_b can be defined, respectively, as

$$Dif_{spatial}(I_a, I_b) = 64\cdot\frac{\Delta\left(flatten\left(I_a^{''}\right), flatten\left(I_b^{''}\right)\right)}{(l\times w\times h)}, \tag{4}$$

$$Dif_{structural}(I_a, I_b) = 64\cdot\frac{count\left(I_a^{''}\ XOR\ I_b^{''}\right)}{(l\times w\times h)}, \tag{5}$$

where *flatten* and *count* denote the flattening and counting operations respectively, and $\Delta(a, b) = (count(a == 1) - count(b == 1))$. In addition, it is evident that $0 \le Dif_{spatial} \le 1$, $0 \le Dif_{structural} \le 1$, and iff $I_a^{''} \equiv I_b^{''}$, $Dif_{spatial},\ Dif_{structural} = 0$.

For convenience, we have presented a visual depiction using three-dimensional patches (note that we are using a slice from each patch as the visualization output) as the illustrative examples. The specific flowchart of the algorithm is delineated in the Fig. 2. It is worth noting that Down-Sampling is performed to optimize resource utilization. The results obtained by computing similarity using the original images are generally similar to those obtained after down-sampling twice. However, there is a significant reduction in time complexity. When Down-Sampling is performed for a third time or more, the similarity may undergo more substantial changes. Additionally, Down-Sampling can help reduce the differences between the same image before and after a short-distance displacement to some extent. Therefore, we have chosen to perform two rounds of Down-Sampling.

Both the spatial and structural information are of equal significance when computing the dissimilarity between two patches. Hence, this article uses the arithmetic mean of $Dif_{spatial}$ and $Dif_{structural}$ to replace the traditional difference calculation method in the Relief model. The new formula for calculating feature weights can be written as follows:

$$w(j) = w(j) - \frac{\mu}{2\cdot k}\sum_{l=1}^{k}\left(Dif_{spatial}\left(x_i, N_l^{+}(x_i)\right) + Dif_{structural}\left(x_i, N_l^{+}(x_i)\right)\right) + \frac{\mu}{2\cdot k}\sum_{l=1}^{k}\left(Dif_{spatial}\left(x_i, N_l^{-}(x_i)\right) + Dif_{structural}\left(x_i, N_l^{-}(x_i)\right)\right). \tag{6}$$

Remark: After preprocessing the MR images corresponding to each patient, we segment the entire image into lots of patches based on their size and assign unique identifiers to each patch. Therefore, there will be no overlapping patches in our patch selection strategy. The detailed flow of the patch pre-selection strategy is shown as follows.

The Patch Pre-selection Strategy

Input: A training dataset D, the number of the nearest neighbor k, the number of samplings s, the maximum number of iterations *max_iter*
Output: The weights of each patch w

```
Initialize the weights w = zeros(n, m).
Divide the training dataset D into D_p and D_n.
For iter = 1 to max_iter
  For i = 1 to s
    Random select a sample x ∈ D.
    Retrieve a sample set N+(x) of k nearest neighbor samples from D_p.
    Retrieve a sample set N−(x) of k nearest neighbor samples from D_n.
    For j = 1 to m
      Update the weight of j-th patch w(j) via Eq. (6).
    End For
  End For
End For
Return The weights of each patch w.
```

3.2. Patch-level global and local attention-based multi-instance deep learning network

After the patch pre-selection module, we have roughly identified the patches that are most likely to contain lesion areas. In order to further improve the model's focus on the specific lesion areas and increase accuracy in diagnosis, two attention mechanism blocks, the inner-patch local attention block and the outer-patch global attention block, are designed in this section.

(a) The inner-patch local attention block

The initial section of this block serves as the underlying structure, which endeavors to acquire highly abstract characteristics from the primary patches and minimize the dimension of feature maps. The proposed underlying structure comprises four layers, each of which is composed of a 3D convolutional module followed by a 3D batch normalization module and a *ReLU* activation function. For clarity, we denote the four layers as $Layer_1$ to $Layer_4$, and the corresponding convolutional modules as $Conv_1$ to $Conv_4$, respectively. The batch normalization and activation functions are referred to as *BN* and *ReLU*, respectively. The kernel size of $Conv_1$ is set to $4\times4\times4$, while $Conv_2$ to $Conv_4$ employ a kernel size of $3\times3\times3$. The channels depths of $Conv_1$ to $Conv_4$ are sequentially set to 12, 36, 48 and 48. In addition, to decrease the size of feature maps and enhance the extraction of significant information from them, we have incorporated a 3D max pooling layer with a filter size of $2\times2\times2$ between $Layer_2$ and $Layer_3$ for down-sampling. Afterwards, we designed an inner-patch local attention mechanism based on channel attention mechanism to enhance the model's attention to important information in the channels. The architecture of the inner-patch local attention block is depicted in Fig. 3. Additionally, assuming the input size of selected patch is $1\times32\times32\times32$, the specific implementation details of $Layer_1$ to $Layer_4$ are shown in Fig. 3(b).

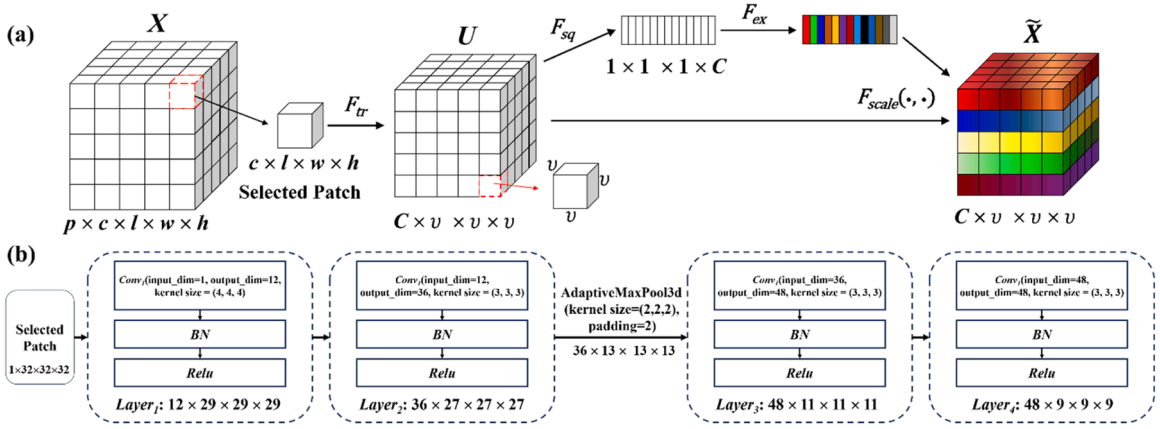


Fig. 3. The architecture of the inner-patch local attention block. (a) Overall architecture; (b) Details implementation of F_{tr}.

Then we present the mathematical model of the designed inner-patch local attention block. Suppose that the training dataset is denoted by $\boldsymbol{D} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \cdots, \boldsymbol{X}_n\}$, where $\boldsymbol{X}_i \in \mathbb{R}^{p \times c \times l \times w \times h}$, and c, l, w, h indicate the number of channels, length, weight, height of the input image, respectively, p is the number of selected patches. Subsequently, we define the feature extraction operation of the underlying structure as F_{tr} with θ_{tr} denoting the corresponding trainable parameters. For any $\boldsymbol{X} \in \boldsymbol{D}$, the output $\boldsymbol{U}$ can be obtained via the following equation:

$$\boldsymbol{U} = F_{tr}(\boldsymbol{X}, \theta_{tr}), \tag{7}$$

where $\boldsymbol{U} = \{\boldsymbol{U}_1, \boldsymbol{U}_2, \cdots, \boldsymbol{U}_C\}$ and $\boldsymbol{U}_i \in \mathbb{R}^{v \times v \times v}$, C is the number of channels. Then, we use a global average pooling operation F_{sq} to produce channel-wise statistics, where the F_{sq} is applied to each channel c and can be expressed as

$$F_{sq}(\boldsymbol{U}_c) = \frac{1}{v \times v \times v} \sum_{i=1}^{v} \sum_{j=1}^{v} \sum_{k=1}^{v} \boldsymbol{U}_c(i,j,k). \tag{8}$$

After that, in order to leverage the information gathered during the squeeze operation, we elect to utilize a basic gating mechanism with a sigmoid activation:

$$F_{ex}\big(F_{sq}(\boldsymbol{U}), \mathbf{W}\big) = ReLU\big(\mathbf{W}_2 \cdot \delta\big(\mathbf{W}_1 \cdot F_{sq}(\boldsymbol{U})\big)\big), \tag{9}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are two trainable weight matrixes. The output of this block is ultimately generated by rescaling the intermediate feature map $\boldsymbol{U}$ with $F_{ex}(F_{sq}(\boldsymbol{U}), \mathbf{W})$:

$$F_{scale}\big(\boldsymbol{U}, F_{ex}\big(F_{sq}(\boldsymbol{U}), \mathbf{W}\big)\big) = F_{ex}\big(F_{sq}(\boldsymbol{U}), \mathbf{W}\big) \cdot \boldsymbol{U}, \tag{10}$$

where $\widetilde{\boldsymbol{X}} = F_{scale}(\boldsymbol{U}, F_{ex}(F_{sq}(\boldsymbol{U}), \mathbf{W})) = \{\widetilde{\boldsymbol{x}}_1, \widetilde{\boldsymbol{x}}_2, \cdots, \widetilde{\boldsymbol{x}}_C\}$, and each $\widetilde{\boldsymbol{x}}_i$ can be calculated by $F_{ex}(F_{sq}(\boldsymbol{U}_i), \mathbf{W}) \cdot \boldsymbol{U}_i$.

(a) The outer-patch global attention block

After the inner-patch local attention block, we apply a mean average operation along the channel dimension for each patch to obtain the final feature representation of each patch. The attention mechanism employed in the outer-patch global attention block is similar to the one used in the local attention block. In this module, we regard each patch as a channel in the local attention mechanism, and introduce a patch-level attention mechanism followed by a feature decoding structure. The decoding structure consists of two convolutional layers with kernel sizes of $2 \times 2 \times 2$, namely $Conv_5$ and $Conv_6$, and two linear layers, namely $Line_1$ and $Line_2$. The channel depths of $Conv_5$ and $Conv_6$ are set to 32 and 16, respectively, and each is followed by batch normalization and *ReLU* activation. Additionally, a global average pooling operation is performed after $Conv_6$. The out features for $Line_1$ and $Line_2$ are set to 16 and 2, respectively. *ReLU* activation is applied after $Line_1$, while Softmax is used after $Line_2$. The detailed structure of the outer-patch global attention block and decoding network is shown in Fig. 4, moreover, suppose

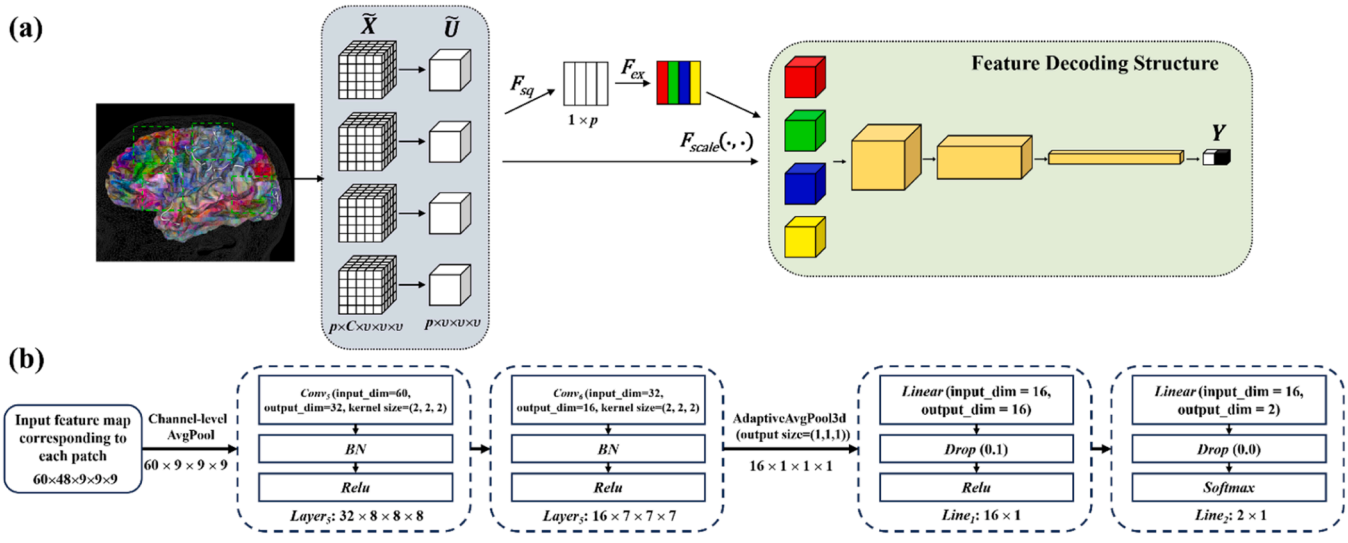


Fig. 4. The architecture of the outer-patch global attention block. (a) Overall architecture; (b) Details implementation of Feature Decoding Structure.

that the input size of feature map is $60 \times 48 \times 9 \times 9 \times 9$, the specific implementation detail of Feature Decoding Structure ($Layer_1$, $Layer_4$, $Line_1$, and $Line_2$) is shown in Fig. 4(b).

Next, we will describe the proposed outer-patch global attention block using a series of mathematical formulas. Assuming that $\widetilde{\boldsymbol{X}} = \{\widetilde{\boldsymbol{X}}_1, \widetilde{\boldsymbol{X}}_2, \cdots, \widetilde{\boldsymbol{X}}_n\}$ is the training set obtained through the inner-patch local attention block, where $\widetilde{\boldsymbol{X}}_i \in \mathbb{R}^{p \times C \times v \times v \times v}$. Then the channel-level average pooling operation can be defined as

$$\widetilde{\boldsymbol{U}} = F_{cap}(\widetilde{\boldsymbol{X}}), \tag{11}$$

where $\widetilde{\boldsymbol{U}} = \{\widetilde{\boldsymbol{U}}_1, \widetilde{\boldsymbol{U}}_2, \cdots, \widetilde{\boldsymbol{U}}_n\}$ and $\widetilde{\boldsymbol{U}}_i \in \mathbb{R}^{p \times v \times v \times v}$. Subsequently, similar to the attention mechanism described above, a novel feature representation is obtained through $F_{scale}(\widetilde{\boldsymbol{U}}, F_{ex}(F_{sq}(\widetilde{\boldsymbol{U}}), \mathbf{W}))$. Then, suppose that the decoding structure is F_{decode}, and the classification result Y can be denoted as

$$Y = F_{decode}\left(F_{scale}\left(\widetilde{\boldsymbol{U}}, F_{ex}\left(F_{sq}(\widetilde{\boldsymbol{U}}), \mathbf{W}\right)\right), \theta_{de}\right), \tag{12}$$

where $Y = \{0, 1\}$ represents the final prediction result, and θ_{de} indicates the trainable parameters of the decoding network. The above-mentioned network (PLGLA) aims at learning an optimal map: $\boldsymbol{X} \rightarrow \boldsymbol{Y}$ by minimizing the cross-entropy loss function.

3.3. Patch-level instance distribution-based active learning strategy

Due to the scarcity of actual MRI samples in real-world scenarios and the difficulty in diagnosis, it is urgent to train a model with high disease recognition rate using the least amount of training data. In the previous section, we have learned the importance of each instance from the outer-patch global attention block. Next, we attempt to design an active learning-based sampling strategy based on the distribution of instances. Specifically, we first use a network framework to implicitly learn the data distribution of high-importance instances, i.e., the mean and variance of the Gaussian distribution. Then, based on the known instance distribution and class information, we design a sample selection strategy. The detailed mathematical model of this strategy is shown below:

Suppose that $D_L = (X^L, Y^L) = \{(x_i^L, y_i^L)\}_{i=1}^{n_L}$ is the labeled dataset, $D_C = X^C = \{x_j^C\}_{j=1}^{n_C}$ is the candidate dataset, where $n_L \ll n_C$, $y_i^L \in \{0,1\}$, $D_L^+ = \{\{(x_i^L, y_i^L) | y_i^L = 1\}_{i=1}^{n_L^+}\} \subseteq D_L$ and $D_L^- = \{\{(x_i^L, y_i^L) | y_i^L = 0\}_{i=1}^{n_L^-}\} \subseteq D_L$. Then, we focus on D_L, and using the PLGLA model, we can compute two instance-level weight matrixes, namely $\mathbf{W}_{D_L}^+ \in \mathbb{R}^{n_L^+ \times p}$ and $\mathbf{W}_{D_L}^- \in \mathbb{R}^{n_L^- \times p}$, for the positive and negative classes, respectively. The mathematical formula is denoted as

$$\mathbf{W}_{D_L}^{\pm} = PLGLA\left(D_L^{\pm}\right). \tag{13}$$

(a) Implicit distributed learning block:

It is obviously that not all the instances are equally important. In this block, we select the top-k most important instances from both the positive and negative classes. Then, we construct a network structure to implicitly learn the data distribution of these important instances. The specific structure of this network includes an encoder module, a decoder module, and a data distribution learning module.

The encoder module is utilized to encode the selected high-importance instances into a latent space, and it comprises of three 3D convolutional layers and a linear layer. The channel depths of these 3D convolutional layers are set to 24, 48, and 72, respectively. The kernel size, stride, and padding are set to $3 \times 3 \times 3$, 2, and 1, respectively. Each of these 3D convolutional layers is followed by a batch normalization (*BN*) layer and rectified linear unit (*ReLU*) activation function. The linear layer has an output size of 384 and is also followed by a *ReLU* activation function.

The decoder module is used to reconstruct the instances from the latent space. In contrast to the encoder module, the decoder module consists of a linear layer and three deconvolutional layers, whose corresponding parameters correspond to those of the encoder module. Except for the last deconvolutional layer, which is followed by a *Tanh* activation function, the structure of the other layers remains unchanged.

The data distribution learning module is used to implicitly learn the data distribution of high-importance instances in the latent space. This module comprises of three linear layers, of which the first two are parallel, aimed to output the mean and variance (μ, σ) of the instance. Subsequently, we perform sampling based on probability, and the sampling formula is given by Eq. (14). The purpose of the third linear layer is to map the low-dimensional representation of the sampled data to a higher dimension.

$$L = \mu + e^{\sigma/2} \cdot eps, \tag{14}$$

where eps denotes a noise vector sampled from the standard normal distribution (with mean 0 and variance 1). During the process of learning the implicit instance distribution, we utilize the similarity between input instances and reconstructed instances as the objective function. The specific architecture is shown in Fig. 5.

(a) Active Learning-based sampling block

In this block, our objective is to select appropriate samples for annotation. As previously described, we have learned the implicit data distribution of each instance, and subsequently, we have designed an active learning sampling strategy based on these distributions. Let D_L, D_C, D_L^+, and D_L^- denote the labeled dataset, candidate dataset, labeled positive dataset, and labeled negative dataset, respectively. Additionally, we assume that the set of the top-k most significant instances in D_L^+, and D_L^- are represented by sets $U_{D_L^+} = \{U_1^+, U_2^+, \cdots, U_k^+\}$ and $U_{D_L^-} = \{U_1^-, U_2^-, \cdots, U_k^-\}$, respectively, and their distributions can be denoted as $p(U_i^+)$ and $p(U_i^-)$. Note that the same instance may be selected in both $U_{D_L^+}$ and $U_{D_L^-}$. After that, for each $X \in D_C$, where $X = \{x_1, x_2, \cdots, x_p\}$, x_i represents the i th instance of X and their distributions can be described as $q(x_i)$, then the uncertainty of X can be defined as

$$amb(X) = \left| \sum_{U_i^+ \in U_{D_L^+}} \int \frac{p(U_i^+) \log p(U_i^+)}{q(x_i)} - \sum_{U_i^- \in U_{D_L^-}} \int \frac{p(U_i^-) \log p(U_i^-)}{q(x_i)} \right|. \tag{15}$$

It can be observed from Eq. (15) that as the value of *amb* increases, the classification of the current sample becomes more certain, whereas a smaller value of *amb* indicates a more ambiguous classification of the current instance X. The detailed flow of the active learning-based sampling strategy is shown as follows.

Active Learning-based Sampling Strategy (PIDAC)

Input: A labeled dataset D_L, a candidate dataset D_C, the number of selected instances k, the max number of iterations max_{iter},
Output: An optimal sample subset R
1. Initialize $R = D_L$ and $iter = 1$.
2. **While** $iter \leq max_{iter}$ and $D_C \neq \varnothing$
3. **For** each sample $x \in D_C$
4. Calculate the uncertainty via Eq. (15).
5. **End For**
6. Let $x_{best} = \underset{x}{\operatorname{argmin}}(amb(x, R, D_C))$.
7. Let $R = R \bigcup x_{best}$.
8. Let $D_C = D_C - \{x_{best}\}$.
9. Let $iter = iter + 1$.
10. **End While**
11. **Return** the optimal sample subset R.

4. Experiment results and analysis

In this section, we will showcase the experimental configurations

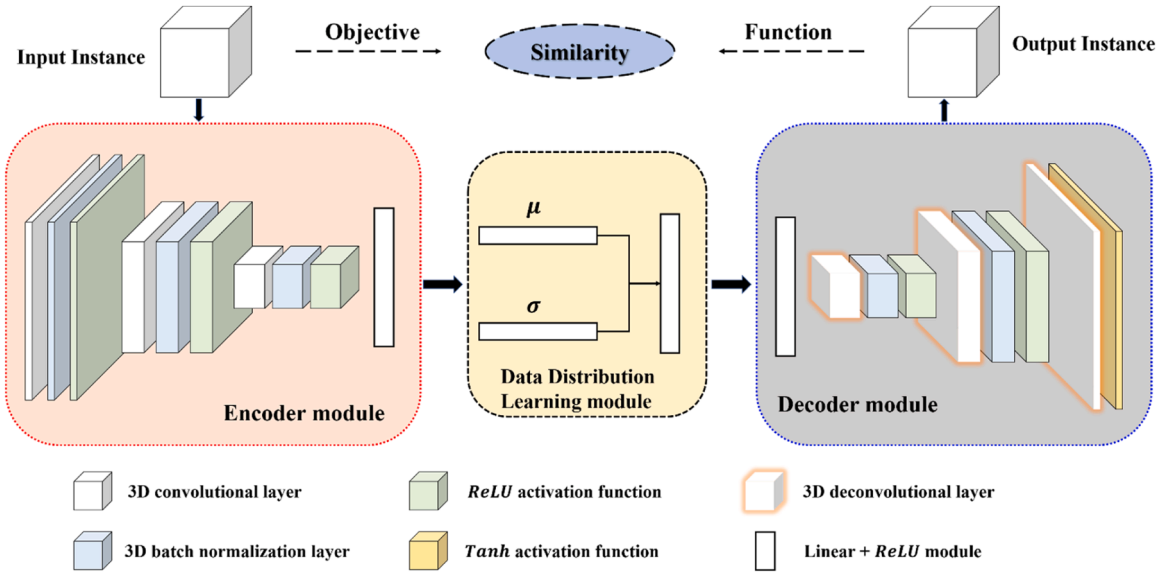


Fig. 5. The architecture of the implicit distributed learning block.

and evaluate the performance and generalizability of our designed classification and active learning methods on various AD-related diagnostic tasks. Our approach will be compared with several state-of-the-art methodologies to demonstrate its effectiveness.

4.1. Dataset and experiment preparation

The Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, which can be obtained from https://adni.loni.usc.edu/, is currently the most extensively researched publicly accessible dataset for AD detection. In this paper, we utilize a dataset consisting of 334 patients with Alzheimer's Disease (AD), 431 patients with Normal Cognition (NC), 258 patients with progressive Mild Cognitive Impairment (pMCI), and 351 patients with stable Mild Cognitive Impairment (sMCI). The objective of this section is to validate the effectiveness of the designed model through the aforementioned dataset. To differentiate between pMCI and sMCI, we establish the criterion that patients who progress from MCI to AD within a period of three years are classified as pMCI, while those who do not show such progression are classified as sMCI. The specific details of the data are presented in Table 1. To facilitate better feature learning and classification, we pre-processed the original structural MRI data downloaded from ADNI. The specific processing process is consistent with the reference [6].

Our model consists of two primary modules, namely PLGLA and PIDAC, each with their unique set of parameters. The PLGLA network was trained using the SGD optimizer with a momentum of 0.9, weight decay of 0.01, and a learning rate of 5e-3. We trained this module for a maximum of 150 epochs, utilizing a batch size of 12 per epoch. On the other hand, we trained the PIDAC network using the Adam optimizer with a first momentum of 0.9, second momentum of 0.999, and learning rate of 1e-3. We set the maximum number of iterations to 100 and used a batch size of 12. The patch number and patch size are ultimately determined to be 60 and $32 \times 32 \times 32$, respectively. In the parameter analysis part, we provide a detailed demonstration of the experimental performance by varying the number and size of patches.

Table 1
The statistical information for subjects enrolled in ADNI.

Class	Male/ Female	Age	MMSE
Alzheimer's Disease (AD)	185/149	74.95±7.82	23.19±2.06
Normal Cognition (NC)	208/205	74.75±5.73	29.06±1.12
stable Mild Cognitive Impairment (sMCI)	215/136	72.36±6.79	27.95±1.80
progressive Mild Cognitive Impairment (pMCI)	153/105	74.12±6.98	26.87±1.75

Values are Reported as Mean ± Standard Deviation.
MMSE: Mini-Mental State Examination.

All experiments in this paper are run on a Windows 10-based personal computer with 32.00 GB of RAM, equipped with an NVIDIA GeForce RTX 3080 Ti GPU and a 3.61 GHz Intel i7–12,700 CPU. All code implementations in this paper were carried out using the PyTorch deep learning framework based on Python 3.9 and PyTorch 1.12.1.

We employ four metrics to assess the classification performance, namely Accuracy ($ACC = (TP + TN)/(TP + FN + TN + FP)$), Specificity ($SPE = TN/(TN + FP)$), Sensitivity ($SEN = TP/(TP + FN)$), and the area under curve (AUC), where TP, TN, FP and FN denote the true positive, true negative, false positive and false negative in the confusion matrix, respectively. Among them, ACC, SPE, and SEN are computed utilizing the default threshold value of 0.5. On the other hand, AUC is determined by examining all possible combinations of true positive rate ($TPR = SEN$) and false positive rate ($FPR = 1 - SPE$), which involves modifying the threshold applied to the prediction outcomes derived from our trained PLGLA network.

4.2. PLGLA compared with other alzheimer's disease detection methods

To showcase the capability of the proposed PLGLA classifier, we conduct a series of controlled experiments, including AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, and sMCI vs. NC, on the ADNI dataset. In these experiments, the performance of the PLGLA method is compared with the eight state-of-the-art detection algorithms, namely VBM [35], ROI [36], PLM [37], DMIL [26], DM^{2}L [38], HFCN [39], HybNet [40], and DA-MIDL [6], in terms of ACC, SPE, SEN, and AUC. All parameters and experimental data of the compared methods can be observed in References [6] and [41]. For the sake of standardization, akin to the References [6] and [41], we train the PLGLA classifier employing a five-fold cross-validation strategy. The specific experimental results are shown in Tables 2 and 3, where the optimal results for each metric are highlighted in bold. In particular, as algorithms DM^{2}L and HybNet were not tested on the pMCI vs. NC and sMCI vs. NC groups, table 3 does not present the specific results for these two algorithms.

As shown in Table 2, for the AD vs. NC and pMCI vs. sMCI pairwise comparison experiments, the PLGLA method achieved the best results in

Table 2
Results of the classification of Alzheimer's Disease (AD)/Normal Control (NC) and progressive Mild Cognitive Impairment (pMCI)/ stable Mild Cognitive Impairment (sMCI) using PLGLA along with other eight compared approaches.

Methods	AD vs. NC				pMCI vs. sMCI			
	ACC	*SEN*	*SPE*	*AUC*	*ACC*	*SEN*	*SPE*	*AUC*
VBM	0.816	0.756	0.875	0.883	0.679	0.629	0.717	0.709
ROI	0.804	0.718	0.888	0.852	0.667	0.571	0.739	0.692
PLM	0.848	0.846	0.850	0.905	0.716	0.657	0.761	0.732
DMIL	0.892	0.859	0.925	0.950	0.765	0.714	0.804	0.790
DM^2L	0.911	0.935	0.881	0.959	0.769	**0.824**	0.421	0.776
HFCN	0.905	0.897	0.913	0.942	0.778	0.686	0.848	0.812
HybNet	0.919	0.824	0.945	**0.965**	0.827	0.579	0.866	0.793
DA-MIDL	0.924	0.910	0.938	**0.965**	0.802	0.771	0.826	**0.851**
PLGLA	**0.967**	**0.955**	**0.976**	**0.965**	**0.828**	0.727	**0.910**	0.819

Table 3
Results of the classification of progressive Mild Cognitive Impairment (pMCI) / Normal Control (NC) and stable Mild Cognitive Impairment (sMCI) / Normal Control (NC) using PLGLA along with other six compared approaches.

Methods	pMCI vs. NC				sMCI vs. NC			
	ACC	*SEN*	*SPE*	*AUC*	*ACC*	*SEN*	*SPE*	*AUC*
VBM	0.816	0.647	0.888	0.853	0.698	0.674	0.713	0.742
ROI	0.789	0.618	0.862	0.846	0.675	0.652	0.688	0.698
PLM	0.825	0.765	0.850	0.876	0.738	0.652	0.788	0.756
DMIL	0.868	0.735	0.925	0.908	0.794	0.783	0.800	0.808
HFCN	0.877	0.795	0.913	0.910	0.802	0.717	**0.850**	0.832
DA-MIDL	**0.895**	**0.824**	0.925	**0.917**	0.825	0.804	0.838	**0.860**
PLGLA	0.868	0.719	**0.962**	0.840	**0.830**	**0.859**	0.809	0.834

the former, particularly with a remarkable *ACC* score of 0.967, significantly outperforming the second-ranked algorithm that attained 0.924. In the latter, PLGLA exhibited superior performance in terms of *ACC* and *SPE*, while slightly falling behind DA-MIDL in terms of *SEN* and *AUC*. As can be seen from Table 3, in the pMCI vs. NC experiment, the PLGLA method achieved a higher *SPE* result and a lower *SEN* result, indicating better performance in classifying pMCI while exhibiting comparatively poorer performance in classifying NC. While this may result in many cases of NC being misdiagnosed as pMCI, the greater harm, from a patient's health perspective, lies in misdiagnosing pMCI as NC [42]. In the sMCI vs. NC experiment, PLGLA obtained the optimal values of 0.830 and 0.859 for the *ACC* and *SEN* metrics, respectively. However, it performed less favorably than HFCN and DA-MIDL in terms of the *SPE* and ranked slightly below DA-MIDL in terms of the *AUC*. Overall, considering all experiments, the performance of PLGLA slightly outperforms the other comparative methods.

4.3. PIDAC compared with other active learning algorithms under classifier PLGLA

To evaluate the effectiveness of the designed sampling strategy PIDAC, this study compares it with three state-of-the-art active learning algorithms: DFAL [43], ASEs [44], and BALD [45], as well as a random sampling strategy: Random Sampling. In this section, we conducted a series of tests in terms of *ACC, AUC, SPE*, and *SEN* with different training sampling ratios (*P*), ranging from 10 % to 90 % with a step size of 10 %. The training set and test set were divided in an 8:2 ratio, and we fixed the number of iterations at 60. The specific experimental results for Tasks 1–4, representing AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, and sMCI vs. NC, respectively, are presented in Tables 4 and 5. Due to space limitations, we only present the specific results of Random Sampling, BALD, and PIDAC in the Tables.

To provide a more visually informative comparison of the

Table 4
ACC and AUC of three active learning algorithms with different values of *P* (%).

Metrics	*P*	Random Sampling				BALD				PIDAC			
		Task1	Task2	Task3	Task4	Task1	Task2	Task3	Task4	Task1	Task2	Task3	Task4
ACC	10 %	0.613	0.541	0.519	0.516	0.740	0.598	0.563	0.601	0.760	0.598	0.644	0.575
	20 %	0.707	0.582	0.600	0.621	0.753	0.607	0.637	0.634	0.787	0.672	0.696	0.680
	30 %	0.753	0.607	0.585	0.673	0.847	0.664	0.748	0.699	0.900	0.697	0.785	0.739
	40 %	0.813	0.689	0.600	0.693	0.840	0.680	0.748	0.716	0.880	0.705	0.733	0.780
	50 %	0.813	0.688	0.719	0.732	0.907	0.689	0.682	0.726	0.913	0.765	0.822	0.824
	60 %	0.873	0.680	0.718	0.791	0.880	0.738	0.733	0.797	0.947	0.787	0.830	0.817
	70 %	0.900	0.773	0.726	0.804	0.913	0.762	0.770	0.804	0.920	0.836	0.852	0.837
	80 %	0.873	0.771	0.756	0.830	0.940	0.795	0.756	0.824	0.940	0.811	0.822	0.876
	90 %	0.880	0.738	0.763	0.843	0.887	0.762	0.822	0.850	0.960	0.830	0.859	0.876
AUC	10 %	0.603	0.480	0.514	0.512	0.742	0.593	0.557	0.606	0.762	0.593	0.618	0.554
	20 %	0.695	0.560	0.597	0.624	0.752	0.606	0.594	0.635	0.783	0.640	0.676	0.681
	30 %	0.745	0.578	0.581	0.669	0.843	0.659	0.734	0.703	0.898	0.658	0.756	0.734
	40 %	0.807	0.660	0.595	0.689	0.838	0.677	0.731	0.716	0.879	0.677	0.710	0.776
	50 %	0.806	0.666	0.715	0.730	0.905	0.685	0.669	0.725	0.911	0.732	0.806	0.819
	60 %	0.868	0.668	0.715	0.794	0.878	0.731	0.718	0.792	0.946	0.784	0.809	0.812
	70 %	0.897	0.828	0.724	0.800	0.911	0.761	0.749	0.800	0.918	0.824	0.836	0.832
	80 %	0.870	0.754	0.753	0.832	0.939	0.791	0.728	0.825	0.938	0.789	0.799	0.874
	90 %	0.878	0.713	0.759	0.841	0.884	0.754	0.807	0.850	0.959	0.818	0.844	0.876

Table 5
SPE and SEN of three active learning algorithms with different values of P (%).

Metrics	P	Random Sampling				BALD				PIDAC			
		Task1	Task2	Task3	Task4	Task1	Task2	Task3	Task4	Task1	Task2	Task3	Task4
SPE	10 %	0.728	0.819	0.725	0.566	0.705	0.703	0.841	0.554	0.718	0.703	0.763	0.723
	20 %	0.840	0.681	0.739	0.591	0.795	0.609	0.825	0.627	0.872	0.857	0.778	0.663
	30 %	0.852	0.736	0.768	0.723	0.936	0.766	0.813	0.663	0.949	0.875	0.901	0.783
	40 %	0.889	0.819	0.812	0.735	0.885	0.750	0.825	0.747	0.910	0.833	0.827	0.795
	50 %	0.888	0.792	0.855	0.759	0.948	0.766	0.738	0.735	0.962	0.829	0.889	0.868
	60 %	0.938	0.736	0.884	0.759	0.936	0.859	0.800	0.855	0.974	0.844	0.913	0.868
	70 %	0.939	0.875	0.783	0.843	0.962	0.797	0.863	0.843	0.962	0.889	0.914	0.892
	80 %	0.914	0.847	0.855	0.807	0.974	0.875	0.900	0.807	0.987	0.922	0.914	0.892
	90 %	0.901	0.847	0.928	0.868	0.948	0.922	0.888	0.843	0.987	0.875	0.925	0.880
SEN	10 %	0.478	0.240	0.303	0.457	0.778	0.483	0.273	0.657	0.806	0.483	0.473	0.386
	20 %	0.551	0.440	0.455	0.657	0.709	0.603	0.364	0.643	0.694	0.423	0.574	0.700
	30 %	0.638	0.420	0.394	0.614	0.750	0.552	0.655	0.743	0.847	0.440	0.611	0.686
	40 %	0.725	0.500	0.379	0.643	0.792	0.603	0.636	0.686	0.847	0.520	0.593	0.757
	50 %	0.724	0.540	0.576	0.700	0.861	0.603	0.600	0.714	0.861	0.635	0.722	0.771
	60 %	0.797	0.600	0.546	0.829	0.819	0.604	0.636	0.729	0.917	0.724	0.704	0.757
	70 %	0.855	0.780	0.667	0.757	0.861	0.724	0.637	0.757	0.875	0.760	0.759	0.771
	80 %	0.826	0.660	0.652	0.857	0.902	0.707	0.546	0.843	0.889	0.655	0.685	0.857
	90 %	0.855	0.580	0.591	0.814	0.819	0.586	0.727	0.857	0.931	0.760	0.764	0.871

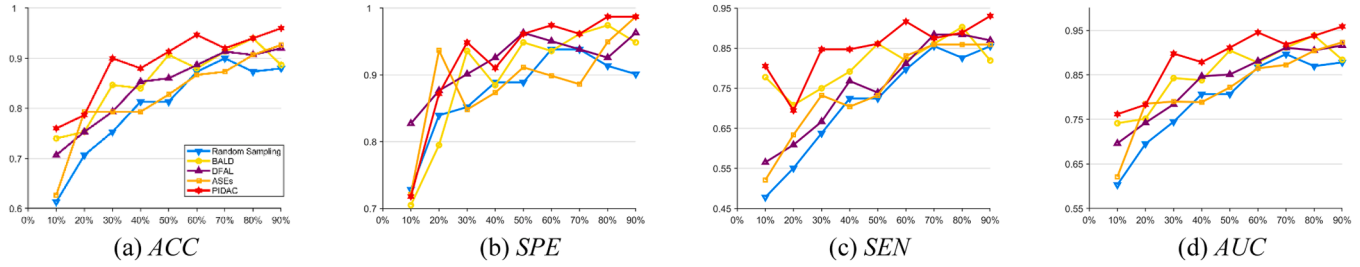


Fig. 6. Comparison results of five sampling methods on Task 1 (AD vs. NC).

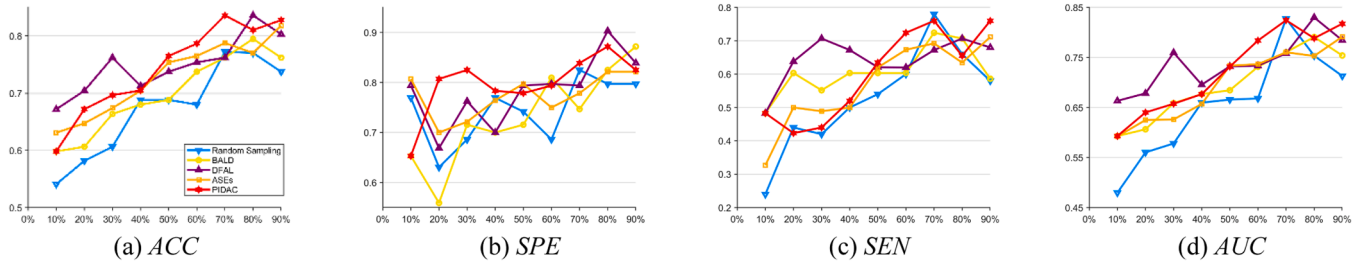


Fig. 7. Comparison results of five sampling methods on Task 2 (pMCI vs. sMCI).

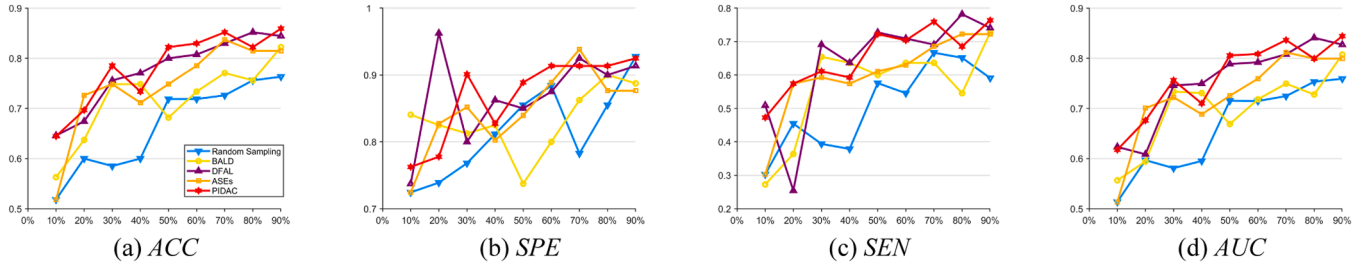
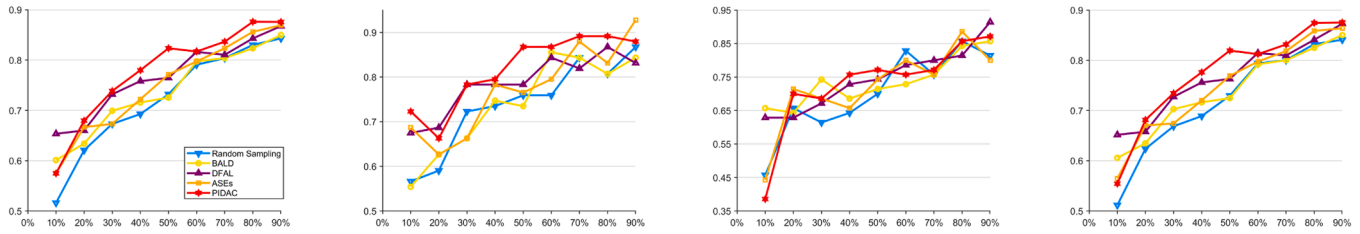


Fig. 8. Comparison results of five sampling methods on Task 3 (pMCI vs. NC).

effectiveness of all five sampling strategies, we generated line graphs depicting the performance metrics for the four tasks. The specific results are presented in the Figs. 6-9, where the X-axis represents the percentage of sampled samples and the Y-axis represents the corresponding values of the metrics.

As can be seen from Tables 4 and 5, due to the variations in the selected samples, the convergence effects of each training iteration also differ. As a result, significant discrepancies in experimental results arise at certain coordinate points. However, our designed PIDAC can select more representative samples for annotation, thereby exhibiting better and faster convergence. It can achieve classification performance close to the optimal results after 60 iterations. In contrast, other sampling



(a) *ACC* (b) *SPE* (c) *SEN* (d) *AUC*

Fig. 9. Comparison results of five sampling methods on Task 4 (sMCI vs. NC).

algorithms, especially Random Sampling strategies, fail to timely select more representative samples for annotation, thus impacting the convergence speed and classification accuracy of the model to varying extents. Since BALD cannot directly process image data, we initially employ a 3D-MLP module to reduce its dimensionality. Subsequently, sampling is conducted on the reduced data. However, this approach may unavoidably result in the loss of critical information. Consequently, BALD may not be applicable for directly sampling image data, particularly when dealing with complex three-dimensional structures.

From Figs. 6-9, it can be observed that for the two deep learning-based active learning methods, DFAL and AEs, their sample selection strategies are more suitable for image data compared to BALD and Random Sampling. Consequently, after selecting a sufficient number of valuable samples for annotation, they can achieve classification performance close to the optimal results at around the 70 % coordinate point. However, our proposed PIDAC approach can achieve the same results at the 50 %−60 % coordinate points. In summary, the sampling strategy based on patch instance distribution, PIDAC, facilitates the selection of more valuable and informative samples for annotation in a faster and earlier manner. As a result, the model can achieve excellent classification performance using fewer samples.

4.4. Ablation study

In this section, we conducted ablation experiments on the designed patch pre-selection module and two attention mechanism modules separately for four comparative tasks: AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, and sMCI vs. NC. The parameters of Patch Number and Patch Size were set to 60 and 32, respectively. Due to the high correlation among the four metrics (*ACC, AUC, SPE*, and *SEN*), in this section, we only used the accuracy metric as the evaluation criterion, and the optimal values for each experimental group are indicated in bold.

For the patch pre-selection strategy, we selected three other patch pre-selection methods for comparative experiments, namely random patch pre-selection (Random Selected), patch pre-selection based on Euclidean distance similarity (Euclidean Distance), and patch pre-selection based on *t*-test (*t*-test). The specific results are shown in Table 6.

For the PLGLA network model, we designed additional network model for comparative experiments, namely PLGLA without attention mechanism (No Attention), PLGLA without global attention mechanism (Local Attention), and PLGLA without local attention mechanism (Global Attention). The specific numerical results are shown in Table 7.

To provide a more intuitive comparison of the results, we designed two line graphs based on the data in Tables 6 and 7. The graphs, shown in Fig. 10, have four points on the horizontal-axis representing the four comparative tasks: AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, and sMCI vs. NC. The vertical-axis represents the accuracy metric for each task.

As can be seen from the above tables and figures, in the process of active learning sampling, the achieved final results are closely related to the pre-selected patches and the weights of the patches output by the global attention mechanism in the network model. Therefore, the aforementioned two sets of ablation experiment also indirectly demonstrate the significant importance of the designed modules in the sampling process.

Table 6
The specific numerical results for four patch pre-selection strategies.

	AD vs. NC	pMCI vs. sMCI	pMCI vs. NC	sMCI vs. NC
Random Selected	0.743	0.698	0.706	0.679
Euclidean Distance	0.921	0.799	0.826	0.811
t-test	0.944	0.812	0.845	0.817
our	**0.967**	**0.828**	**0.868**	**0.830**

Table 7
The specific numerical results of PLGLA and its three variants.

	AD vs. NC	pMCI vs. sMCI	pMCI vs. NC	sMCI vs. NC
No Attention	0.921	0.802	0.844	0.809
Local Attention	0.933	0.807	0.850	0.810
Global Attention	0.958	0.816	0.861	0.828
PLGLA	**0.967**	**0.828**	**0.868**	**0.830**

4.5. Parametric analyses

In our model, two crucial parameters, namely the Patch Number and the Patch Size, require manual configuration. Hence, this section presents the final classification results obtained by experimenting with different parameter settings. Specifically, the Patch Size is selected from the set $\{24 \times 24 \times 24,\ 32 \times 32 \times 32,\ 40 \times 40 \times 40\}$, and the Patch Number is chosen from the set $\{45,\ 60,\ 75\}$. The specific numerical outcomes are provided in Table 8.

For better visualization, we generated four bar charts based on the aforementioned table, where Tasks 1–4 represent AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, and sMCI vs. NC, respectively. Moreover, to enhance the comparability of metrics, we performed a normalization operation on each metric in the bar chart, scaling all results to the range of $[0.1, 1]$. Please refer to Fig. 11 for specific details. The X-axis is represented by a pair of parameters in the form of (Batch Number, Batch Size), and the Y-axis represents four metrics.

Based on the comprehensive results shown in the Table 8 and Fig. 11, selecting a large number or size of patches may result in the inclusion of a significant amount of irrelevant information, while choosing too few or small patches may lead to the loss of crucial information. Only by selecting an appropriate number and size of patches can a stable experimental outcome be consistently maintained. Therefore, this study ultimately selects a Patch Number of 60 and a Patch Size of $32 \times 32 \times 32$.

5. Discussion

In this section, we further elaborate on the model designed in this paper from four perspectives: statistical analysis, efficiency, visualization, and generality.

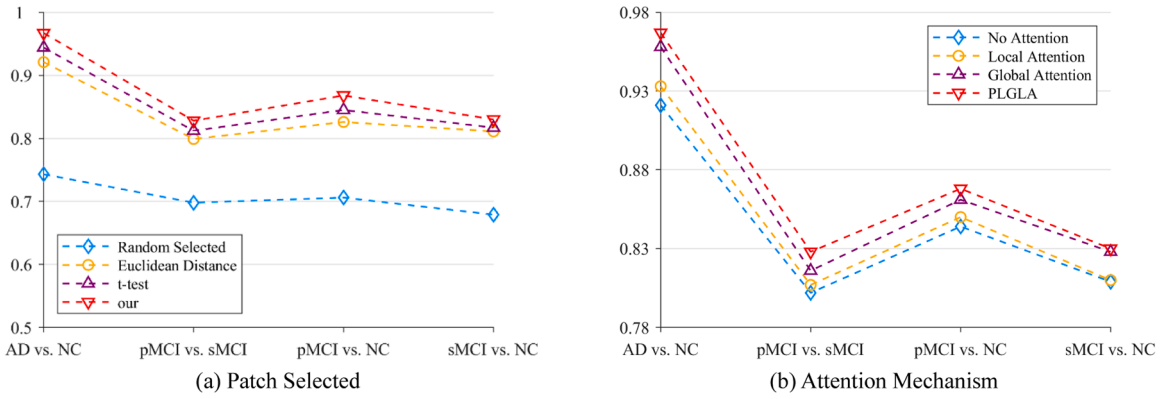


Fig. 10. Comparative analysis of two ablation studies across four tasks.

Table 8
The numerical results with different parameter settings.

Patch Size		24			32			40		
Patch Number		45	60	75	45	60	75	45	60	75
AD	ACC	0.897	0.865	0.860	0.933	**0.967**	0.953	0.947	0.961	0.952
vs.	SEN	0.887	0.797	0.824	0.899	0.955	0.922	0.922	**0.969**	0.952
NC	SPE	0.911	0.921	0.890	0.972	**0.976**	0.977	0.965	0.954	0.954
	AUC	0.899	0.859	0.857	0.935	**0.965**	0.949	0.944	0.961	0.953
pMCI	ACC	0.689	0.702	0.747	0.730	**0.828**	0.771	0.697	0.697	0.787
vs.	SEN	0.380	0.296	0.553	0.548	**0.727**	0.714	0.471	0.500	0.630
sMCI	SPE	0.798	0.848	0.785	**0.917**	0.910	0.818	0.859	0.853	0.912
	AUC	0.589	0.572	0.669	0.733	**0.819**	0.766	0.666	0.677	0.771
pMCI	ACC	0.785	0.861	0.770	0.822	**0.868**	0.854	0.800	0.824	0.852
vs.	SEN	0.620	0.732	0.655	0.685	**0.962**	0.943	0.645	0.709	0.938
NC	SPE	0.882	**0.937**	0.850	0.914	0.719	0.660	0.932	0.900	0.727
	AUC	0.751	0.834	0.752	0.799	**0.840**	0.801	0.788	0.805	0.832
sMCI	ACC	0.771	0.719	0.771	0.821	**0.830**	0.811	0.669	0.613	0.688
vs.	SEN	0.730	0.652	0.730	0.737	**0.859**	0.685	0.354	0.194	0.414
NC	SPE	0.810	0.774	0.810	0.896	0.809	0.925	0.875	**0.951**	0.892
	AUC	0.770	0.713	0.770	0.817	**0.834**	0.805	0.614	0.573	0.653

5.1. Statistical analysis

In this subsection, the Friedman test [46] is used to demonstrate the statistically performance of all the methods under different evaluation metrics (detailed results are illustrated in Tables 2-5, 8). Suppose we compare the performance of K kinds of algorithms on N datasets or tasks, the formula for the Friedman's test is as follows:

$$\tau_{\chi^2} = \frac{12N}{K(K+1)}\left(\sum_{j=1}^{K} R_j^2 - \frac{K(K+1)^2}{4}\right) \tag{16}$$

$$\tau_F = \frac{(N-1)\tau_{\chi^2}}{N(k-1)-\tau_{\chi^2}} \tag{17}$$

where R_j represents the average ranking of the j-th algorithm, and τ_F follows the F distribution with degrees of freedom $K-1$ and $(K-1)(N-1)$. If τ_F exceeds the critical value from the F distribution with degrees of freedom $K-1$ and $(K-1)(N-1)$, it indicates that the assumption of equal performance among all algorithms is rejected. In this scenario, we employ the Nemenyi test [47] to further distinguish each method. Suppose that performance of K kinds of algorithms is compared on N datasets or tasks and the significance level $\alpha = 0.1$, the critical value of the Nemenyi test is expressed as $CD = q_\alpha\sqrt{\frac{K(K+1)}{6N}}$, where q_α indicates the critical value of the Tukey distribution. If the difference between the average ranks of two algorithms exceeds the critical value CD, then the hypothesis of "equal performance between the two algorithms" is rejected with the corresponding confidence level.

Based on the *ACC, SEN, SPN* and *AUC* metrics for the four kinds of binary symptom diagnose tasks (AD vs. NC, pMCI vs. sMCI, pMCI vs. NC, sMCI vs. NC) in Tables 2 and 3, Table 9 displays the average rankings of the nine methods across the four different indicators for each task. It should be noted that due to the absence of results for the DM^{2}L and HybNet algorithms in the pMCI vs. NC and sMCI vs. NC tasks, these two algorithms are assigned a joint rank of last place for the computation of rankings. From Table 9, we can calculate the τ_F values for four indicators as 5.6627, 5.0362, 6.0732 and 4.1287, respectively. Sinc all the τ_F values are greater than $F(8, 24) = 1.940$, it indicates that the performances of the nine algorithms in Tables 2 and 3 are significantly different. We present the Nemenyi test results of the nine algorithms for the four indicators in an intuitive manner, as depicted in Fig. 12. If there is no line with the same color between the two methods in the figure, it suggests that the average ranking difference between the two methods exceeds the critical difference (CD), thus the performance of the top-ranked method is significantly better than that of the lower-ranked method. Fig. 12 indicates that PLGLA performs significantly better than ROI in terms of *ACC* and significantly better than DM^{2}L in terms of *SPE*.

Based on the comparisons of PIDAC with four other active learning methods, Table 10 summarizes the average ranking of all algorithms across each metric. For convenience, we utilize the average values of each metric across the selected sample size range from 10 % to 90 %, as presented in Tables 4 and 5, as the criterion for ranking. On the basis of the results in Table 10, we compute the τ_F values of four metrics are 15.4615, 8.4286, 31.2857 and 15.4615, respectively. Because all the τ_F values are greater than $F(4, 12) = 2.480$, there are significant differences among the five algorithms. Fig. 13 provides a visual

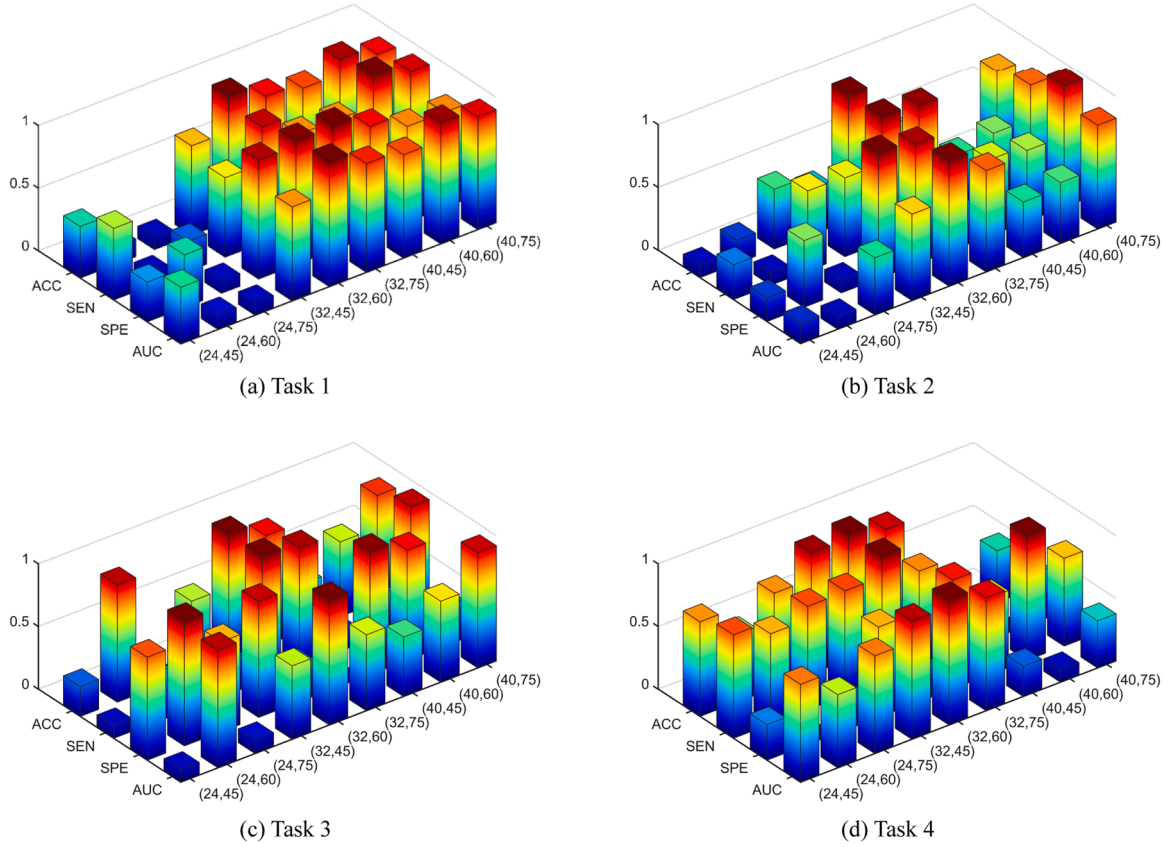


Fig. 11. The visualization of the parameters (Patch Number, Patch Size) on ADNI dataset.

Table 9
Average rankings of nine algorithms on four tasks for *ACC, SEN, SPN*, and *AUC* metrics.

	VBM	ROI	PLM	DMIL	DM^2L	HFCN	HybNet	DA-MIDL	PLGLA
ACC	7	8	6	4.88	6.5	3.5	5.5	2	1.63
SEN	6.5	7.88	5.38	4	5	3.75	8	2	2.5
SPE	6.75	6.5	6.75	3.88	8.25	3.25	5.25	2.88	1.5
AUC	6.75	7.75	5.75	4.25	6.75	3.5	5.75	1.25	3.25

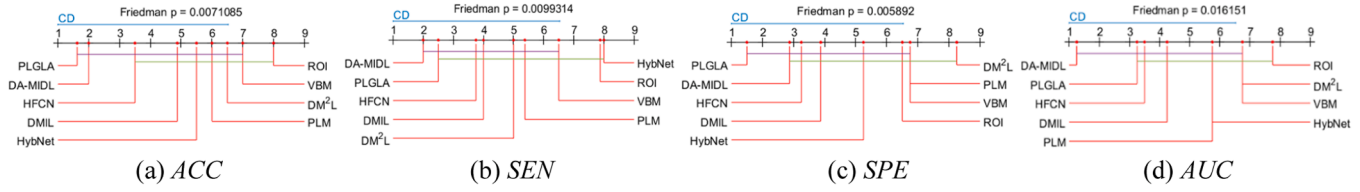


Fig. 12. Statistical analysis of nine classification methods across four metrics.

Table 10
Average rankings of five active learning algorithms on four tasks for *ACC, SEN, SPN*, and *AUC* metrics.

	Random Sampling	BALD	DFAL	ASEs	PIDAC
ACC	5	3.5	2	3.25	1.25
SEN	5	2.5	1.75	3.75	2
SPE	4.75	4	2	3.25	1
AUC	5	3.5	2	3.25	1.25

representation of the Nemenyi test results for the five algorithms across the four metrics. It is evident that PIDAC significantly outperforms Random Sampling in all metrics and exhibits significant superiority over BALD in the *SPE* metric.

Similarly, Table 11 presents the ranking results of the nine parameter combinations from Table 8 across the four metrics. By calculating the τ_F values for *ACC, SEN, SPE*, and *AUC*, we obtain them to be 3.5011, 1.5212, 2.8537, and 3.4358, respectively. It is noteworthy that a situation arises where the τ_F value for *SPE* is less than the critical value $F(8, 24) = 1.940$. Therefore, in our study, we observe no significant differences in the *SEN* metric among the discussed nine parameter combinations, while significant differences exist in the other metrics. A more intuitive Nemenyi test results are illustrated in Fig. 14.

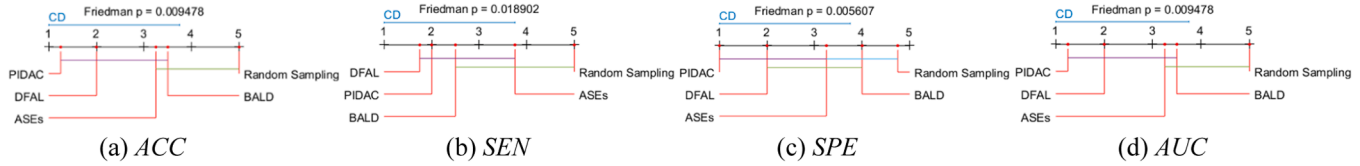


Fig. 13. Statistical analysis of five active learning methods across four metrics.

Table 11
Average rankings of nine kinds of parameters on four tasks for ACC, SEN, SPN, and AUC metrics.

	(24, 45)	(24, 60)	(24, 75)	(32, 45)	(32, 60)	(32, 75)	(40, 45)	(40, 60)	(40, 75)
ACC	7.125	5.5	6.625	4.75	1	3	6.875	5.875	4.25
SEN	6.875	5.75	7.625	2.5	5	5	3.75	3.875	4.625
SPE	6.875	7	5.625	4.75	1.25	3.375	6.875	5.25	4
AUC	7.125	6.25	6.875	4.5	1	3.75	6.75	5	3.75

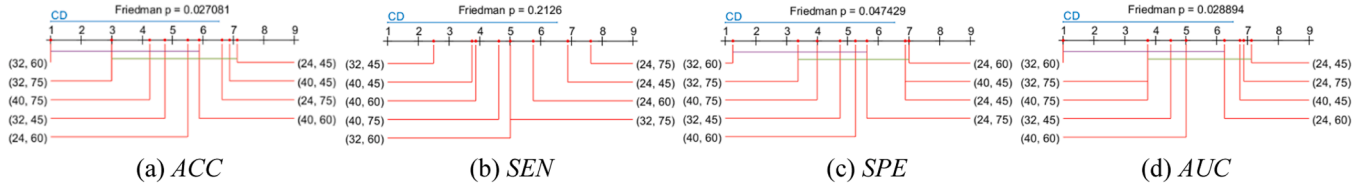


Fig. 14. Statistical analysis of nine parameter combinations across four metrics.

5.2. The efficiency of proposed methods

Considering the significance of computational speed in the field of medical image diagnostics, in this subsection, we conducted an analysis of both time complexity and model complexity for patch pre-selection and the PLGLA model, respectively. In Section 3.1, assuming that the provided ADNI dataset comprises n samples, with each MR image being divided into m patches, and each patch potentially containing f pixels, we first compute the distances between all paired samples based on the hash similarity between blocks, so the approximate time complexity of this step is $O(n{\cdot}m{\cdot}f)$. Next, we proceed with updating the weights for each patch. Given that we need to consider all other samples within the k-nearest neighbor vicinity for each sample, the time complexity of feature weight updating can be expressed as $O(k{\cdot}n{\cdot}m)$. Furthermore, as our patch pre-selection module employs the principle of sampling detection, we repeat the sampling process for a maximum of max_iter iterations. In each iteration, we select s ($s < n$) samples for weight updating. Consequently, the overall time complexity of the designed patch pre-selection can be expressed as $O(n{\cdot}m{\cdot}f) + O(max\_iter{\cdot}k{\cdot}s{\cdot}m)$.

In practical applications, it is common for max_iter and k to be small constants, typically not exceeding 10. Therefore, we can derive $O(max\_iter{\cdot}k{\cdot}s{\cdot}m) < O(n{\cdot}m{\cdot}f)$, whereas the time complexity $O(n{\cdot}m{\cdot}f)$ for computing the distances between patches is an unavoidable time cost in all existing patch pre-selection strategies. Hence, the time spent on the patch pre-selection strategy designed in Section 3.1 is comparable to that of other existing patch pre-selection approaches. Moreover, it is important to note that the patch pre-selection strategy can be performed offline, i.e., The selected patches can be obtained before starting the model training process, without incurring any additional time cost during the training or inference processes.

Table 12
Comparative analysis of PLGLA and DA-MIDL models in terms of Model Size, Parameters, and Flops.

Model	Model Size (*KB*)	Parameters ($\times 10^6$)	Flops ($\times 10^9$)
DA-MIDL	3366	0.86	40.7
PLGLA	**931**	**0.23**	**21.7**

In Section 4.2, we have validated the improved accuracy of PLGLA model, next, to further evaluate its computational resource efficiency, we selected the DA-MIDL model, which has the closest performance to PLGLA, as a comparison. Our comparative analysis encompassed three key aspects: model size, number of parameters, and floating-point operations. The specific comparative results are shown in Table 12. It can be observed that the PLGLA model demonstrated a model size that was two-thirds smaller than that of DA-MIDL, a reduction of three-quarters in parameter count, and approximately half the number of floating-point operations. Consequently, PLGLA achieves higher classification accuracy while operating in a more resource-efficient manner.

5.3. Visualization

In this section, we performed visualization on patients with Alzheimer's Disease (AD) and Mild Cognitive Impairment (MCI). The specific results are shown in Fig. 15. The left half of the figure displays three AD patients, while the right half shows three MCI patients. Considering the limitations of visualizing 3D images, in this figure, we sliced the images from different perspectives (axial, coronal, sagittal) and demonstrated the relative important patches and relative important microstructures within each slice. In Fig. 15, the regions annotated with red boxes are considered the most crucial locations by our model. We can observe that the informative areas, indicated by the red color, are predominantly located at the edges of the sulcus gyrus and gray matter. These regions may effectively reflect local structural changes caused by brain atrophy. Furthermore, the probable pathological locations in AD classification and MCI conversion prediction exhibit significant similarity, which aligns with the high correlation between the two classification tasks based on the progression of Alzheimer's disease. In conclusion, we believe that the proposed approach can assist brain science experts to some extent in the diagnosis of AD, thereby reducing misdiagnosis rates.

5.4. Generalization on AIBL dataset

To substantiate the robustness and generalizability of our model, we extend our evaluation to an independent AIBL dataset (https://aibl.

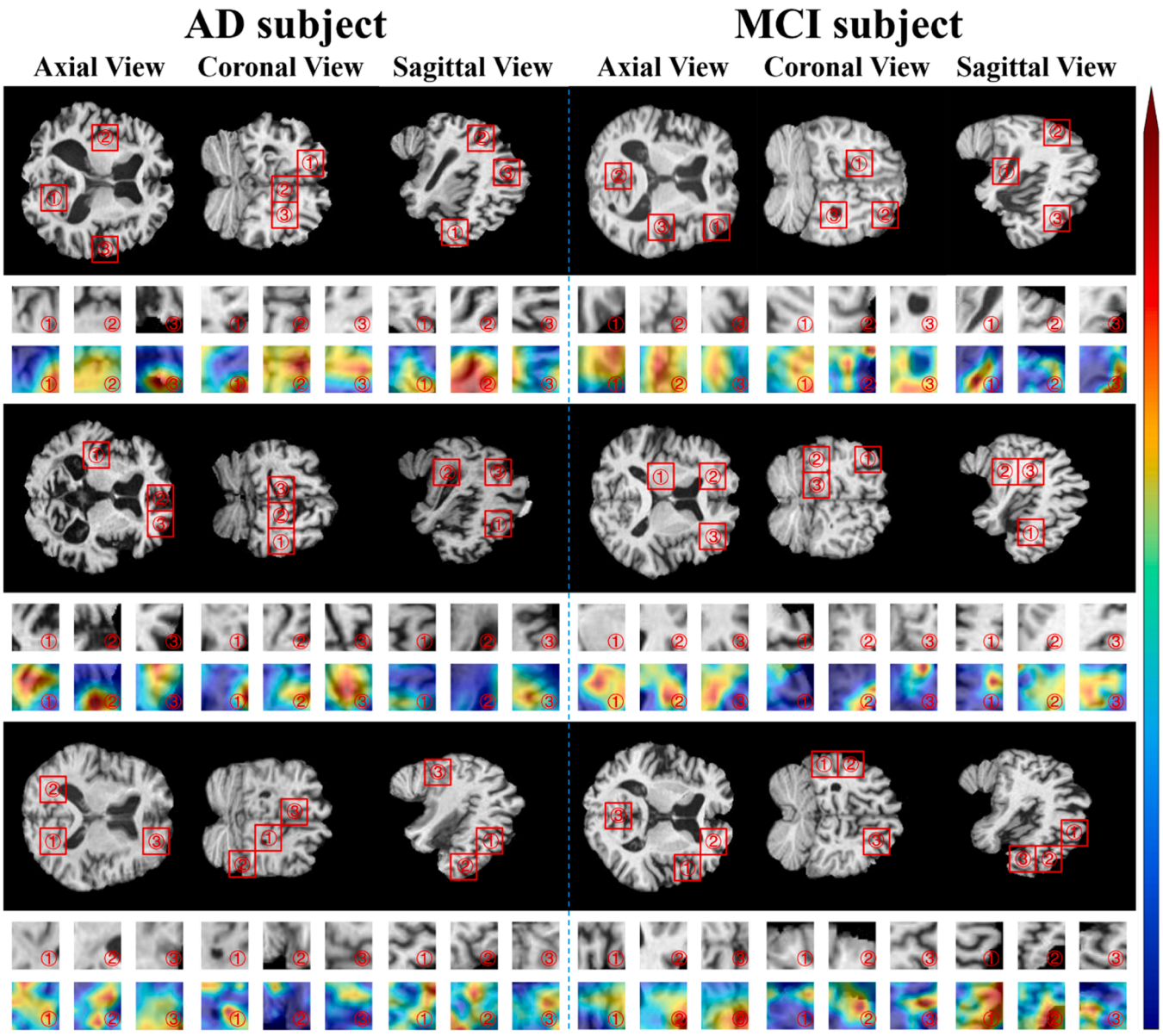


Fig. 15. The visualization of AD and MCI subjects.

Table 13
Results of AD classification and MCI conversion prediction using PLGLA along with other six compared approaches on the AIBL dataset.

Methods	AD vs. NC				pMCI vs. sMCI			
	ACC	*SEN*	*SPE*	*AUC*	*ACC*	*SEN*	*SPE*	*AUC*
VBM	0.808	0.582	0.866	0.817	0.673	0.529	0.720	0.717
ROI	0.793	0.519	0.863	0.796	0.664	0.471	0.710	0.671
PLM	0.839	0.722	0.870	0.846	0.709	0.529	0.742	0.725
DMIL	0.868	0.772	0.893	0.901	0.764	0.588	0.796	0.793
HFCN	0.889	0.823	0.906	0.930	0.782	0.647	0.806	0.796
DA-MIDL	0.902	0.848	0.915	0.939	0.809	0.706	**0.828**	0.824
PLGLA	**0.923**	**0.910**	**0.929**	**0.940**	**0.813**	**0.739**	0.807	**0.833**

csiro.au). This evaluation encompasses the assessment of our PLGLA method as well as its competing methods, all of which were trained on the ADNI dataset. The AIBL dataset utilized in our study comprises 100 samples of AD, 198 samples of NC, 23 samples of pMCI, and 62 samples of sMCI. The specific experimental results for AD classification and MCI conversion prediction on the AIBL dataset are presented in Table 13. All numerical values are derived from reference [6]. As observed in Table 13, our model continues to maintain a leading position in the majority of cases within the AIBL dataset. Therefore, these results demonstrate the sound generalization capability of our model for AD diagnosis.

6. Conclusion

In this study, we propose a patch-based multi-instance Alzheimer's Disease detection network that incorporates a global and local attention mechanism, along with a novel active learning strategy based on the instance distribution, which consists of three major components: (1) A patch pre-selection module was designed to identify regions that are more prone to pathology, while excluding a large amount of background areas and irrelevant information for further investigation. (2) An inner-patch local attention block and an outer-patch global attention block were developed to assist the network model in extracting more discriminative local and global information. (3) An active learning sampling strategy was devised to reduce the cost of data acquisition and expert annotation in real-world scenarios. Through four comprehensive experiments on the ADNI dataset, we have validated the effectiveness of our proposed network framework and active learning strategy from multiple perspectives.

The two proposed models in this study, PLGLA and PIDAC, are designed to achieve higher accuracy in Alzheimer's disease diagnosis while utilizing lower resources and reducing annotation costs. The PLGLA model achieves higher accuracy compared to suboptimal models while reducing computational resource and space consumption by approximately half. The PIDAC method effectively reduces annotation costs by 30 % or even higher.

However, there are still some remaining issues that need to be addressed, which can be outlined as follows. The patch preselection module we designed is based on the Relief algorithm, which introduces a certain level of randomness in the patch selection process. Consequently, when randomly selecting samples of poor quality, it may impact the weight scores of critical patches. Furthermore, our model only considers the MRI modality in the ADNI dataset, neglecting other modalities such as DTI and PET. This limitation may constrain further improvements in model performance. Therefore, in future work, we aim to address these issues and conduct research focusing on the multi-modal domain.

CRediT authorship contribution statement

Tianxiang Wang: Conceptualization, Methodology, Validation, Writing – original draft. **Qun Dai:** Formal analysis, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my data in the body of the paper.

Acknowledgements

This work was supported by the National Key R&D Program of China (Grant Nos. 2018YFC2001600, 2018YFC2001602), and the National Natural Science Foundation of China under Grant No. 61473150.

References

[1] J. Hu, Q. Chen, H.R. Zhu, L.C. Hou, W. Liu, Q.H. Yang, H.D. Shen, G.L. Chai, B. X. Zhang, S.X. Chen, Microglial Piezo1 senses Aβ fibril stiffness to restrict Alzheimer's disease, Neuron 111 (1) (2023) 15–29, e8.
[2] A.C. Yang, R.T. Vest, F. Kern, D.P. Lee, M. Agam, C.A. Maat, P.M. Losada, M. B. Chen, N. Schaum, N. Khoury, A human brain vascular atlas reveals diverse mediators of Alzheimer's risk, Nature 603 (7903) (2022) 885–892.
[3] L. Yu, W. Xiang, J. Fang, Y.P.P. Chen, R.F. Zhu, A novel explainable neural network for Alzheimer's disease diagnosis, Pattern Recognit. 131 (2022).
[4] Q. Zhu, B.L. Xu, J.S. Huang, H.Y. Wang, R.T. Xu, W. Shao, D.Q. Zhang, Deep multi-modal discriminative and interpretability network for Alzheimer's Disease diagnosis, IEEE Trans. Med. Imaging (2022).
[5] T.K.M. Wang, C. Ayoub, M. Chetrit, D.H. Kwon, C.L. Jellis, P.C. Cremer, M. A. Bolen, S.D. Flamm, A.L. Klein, Cardiac magnetic resonance imaging techniques and applications for pericardial diseases, Circulation: Cardiovasc. Imaging 15 (7) (2022) e014283.
[6] W.Y. Zhu, L. Sun, J.S. Huang, L.X. Han, D.Q. Zhang, Dual attention multi-instance deep learning for Alzheimer's disease diagnosis with structural MRI, IEEE Trans. Med. Imaging 40 (9) (2021) 2354–2366.
[7] Q. Yang, X. Li, X.Y. Ding, F.Y. Xu, Z.H. Ling, Deep learning-based speech analysis for Alzheimer's disease detection: a literature review, Alzheimers Res. Ther. 14 (1) (2022) 1–16.
[8] M. Vounou, E. Janousova, R. Wolz, J.L. Stein, P.M. Thompson, D. Rueckert, G. Montana, A.s.D.N. Initiative, Sparse reduced-rank regression detects genetic associations with voxel-wise longitudinal phenotypes in Alzheimer's disease, Neuroimage 60 (1) (2012) 700–716.
[9] W. Shao, Y. Peng, C. Zu, M.L. Wang, D.Q. Zhang, Hypergraph based multi-task feature selection for multimodal classification of Alzheimer's disease, Comput. Med. Imaging Graph. 80 (2020) 101663.
[10] L. Chen, H.Z. Qiao, F. Zhu, Alzheimer's disease diagnosis with brain structural MRI using Multiview-slice attention and 3D convolution neural network, Front. Aging Neurosci. 14 (2022).
[11] M.A. Ganaie, M. Hu, A.K. Malik, M. Tanveer, P. Suganthan, Ensemble deep learning: a review, Eng. Appl. Artif. Intell. 115 (2022) 105151.
[12] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning-based text classification: a comprehensive review, ACM Comput. Surv. (CSUR) 54 (3) (2021) 1–40.
[13] Z.Y. Ren, S.H. Wang, Y.D. Zhang, Weakly supervised machine learning, CAAI Trans. Intell. Technol. (2023).
[14] D.W. Zhang, J.W. Han, G. Cheng, M.H. Yang, Weakly supervised object localization and detection: a survey, IEEE Trans. Pattern Anal. Mach. Intell. 44 (9) (2021) 5866–5885.
[15] J.H. Cai, J. Hao, H.F. Yang, X.J. Zhao, Y.Q. Yang, A review on semi-supervised clustering, Inf. Sci. (Ny) (2023).
[16] W.H. Li, R.Y. Huang, J.P. Li, Y.X. Liao, Z.Y. Chen, G.L. He, R.Q. Yan, K. Gryllias, A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: theories, applications and challenges, Mech. Syst. Signal Process. 167 (2022) 108487.
[17] P. Chlap, H. Min, N. Vandenberg, J. Dowling, L. Holloway, A. Haworth, A review of medical image data augmentation techniques for deep learning applications, J. Med. Imaging Radiat. Oncol. 65 (5) (2021) 545–563.
[18] P.Z. Ren, Y. Xiao, X.J. Chang, P.Y. Huang, Z.H. Li, B.B. Gupta, X.J. Chen, X. Wang, A survey of deep active learning, ACM Comput. Surv. (CSUR) 54 (9) (2021) 1–40.
[19] Y.F. Li, L.Z. Guo, Z.H. Zhou, Towards safe weakly supervised learning, IEEE Trans. Pattern Anal. Mach. Intell. 43 (1) (2019) 334–346.
[20] Z.Y. Dai, J.J. Yi, L. Yan, Q.W. Xu, L. Hu, Q. Zhang, J.H. Li, G.Q. Wang, PFEMed: few-shot medical image classification using prior guided feature enhancement, Pattern Recognit. 134 (2023).
[21] R. Ye, Q. Dai, Implementing transfer learning across different datasets for time series forecasting, Pattern Recognit. 109 (2021).
[22] C.Y. Wang, J.J. Jiang, X. Zhou, X.M. Liu, ReSmooth: detecting and utilizing OOD samples when training with data augmentation, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[23] X.P. Kang, D.Y. Li, S.G. Wang, A multi-instance ensemble learning model based on concept lattice, Knowl. Based Syst. 24 (8) (2011) 1203–1213.
[24] T. Yuan, H. Wenning, J. Dawei, C. Gang, Z. Ao, A review of latest multi-instance learning, in: CSAI 2020: Proceedings of the 2020 4th International Conference on Computer Science and Artificial Intelligence, 2020, pp. 41–45, 11.
[25] Y. Yang, Y.L. Tu, H.C. Lei, W. Long, HAMIL: hierarchical aggregation-based multi-instance learning for microscopy image classification, Pattern Recognit. 136 (2023).
[26] M.X. Liu, J. Zhang, E. Adeli, D.G. Shen, Landmark-based deep multi-instance learning for brain disease diagnosis, Med. Image Anal. 43 (2018) 157–168.
[27] H.D. Couture, J.S. Marron, C.M. Perou, M.A. Troester, and M. Niethammer, "Multiple instance learning for heterogeneous images: training a CNN for histopathology." pp. 254–262, 2018.
[28] C. Barata, M.E. Celebi, J.S. Marques, Explainable skin lesion diagnosis using taxonomies, Pattern Recognit 110 (2021).
[29] X. Chen, T.S. Kuang, H. Deng, S.H. Fung, J. Gateno, J.J. Xia, P.T. Yap, Dual Adversarial attention mechanism for unsupervised domain adaptive medical image segmentation, IEEE Trans. Med. Imaging 41 (11) (2022) 3445–3453.
[30] J.R. Shang, X. Zhang, G.S. Zhang, W.H. Song, J.Y. Chen, Q.L. Li, M.L. Gao, Gated multi-attention feedback network for medical image super-resolution, Electronics (Basel) 11 (21) (2022) 3554.
[31] X.M. Liu, Q. Yuan, Y.Z. Gao, K.L. He, S. Wang, X. Tang, J.S. Tang, D.G. Shen, Weakly supervised segmentation of COVID19 infection with scribble annotation on CT images, Pattern Recognit. 122 (2022).
[32] J. Zhao, S.L. Sun, H.J. Wang, Z.H. Cao, Promoting active learning with mixtures of Gaussian processes, Knowl. Based Syst. 188 (2020) 105044.
[33] B. Gu, Z. Zhai, C. Deng, H. Huang, Efficient active learning by querying discriminative and representative samples and fully exploiting unlabeled data, IEEE Trans. Neural Netw. Learn. Syst. 32 (9) (2020) 4111–4122.

[34] L. Sun, T.Y. Yin, W.P. Ding, Y.H. Qian, J.C. Xu, Multilabel feature selection using ML-ReliefF and neighborhood mutual information for multilabel neighborhood decision systems, Inf. Sci. (Ny) 537 (2020) 401–424.
[35] J. Ashburner, K.J. Friston, Voxel-based morphometry—the methods, Neuroimage 11 (6) (2000) 805–821.
[36] D.Q. Zhang, Y.P. Wang, L.P. Zhou, H. Yuan, D.G. Shen, Multimodal classification of Alzheimer's disease and mild cognitive impairment, Neuroimage 55 (3) (2011) 856–867.
[37] M.H. Liu, D.Q. Zhang, D.G. Shen, Ensemble sparse classification of Alzheimer's disease, Neuroimage 60 (2) (2012) 1106–1116.
[38] M.X. Liu, J. Zhang, E. Adeli, D.G. Shen, Joint classification and regression via deep multi-task multi-channel learning for Alzheimer's disease diagnosis, IEEE Trans. Biomed. Eng. 66 (5) (May 2019) 1195–1206.
[39] C.F. Lian, M.X. Liu, J. Zhang, D.G. Shen, Hierarchical fully convolutional network for joint atrophy localization and Alzheimer's Disease diagnosis using Structural MRI, IEEE Trans. Pattern Anal. Mach. Intell. 42 (4) (Apr, 2020) 880–893.
[40] C.F. Lian, M.X. Liu, Y.S. Pan, D.G. Shen, Attention-guided hybrid network for dementia diagnosis with structural MR images, IEEE Trans. Cybern. 52 (4) (Apr, 2022) 1992–2003.
[41] M. Ashtari-Majlan, A. Seifi, M.M. Dehshibi, A multi-stream convolutional neural network for classification of progressive mci in Alzheimer's disease using structural mri images, IEEE J. Biomed. Health Inform. 26 (8) (2022) 3918–3926.
[42] C.A. Hunter, N.Y. Kirson, U. Desai, A.K.G. Cummings, D.E. Faries, H.G. Birnbaum, Medical costs of Alzheimer's disease misdiagnosis among US Medicare beneficiaries, Alzheimers Dementia 11 (8) (2015) 887–895.
[43] M. Ducoffe, and F. Precioso, "Adversarial active learning for deep networks: a margin based approach," *arXiv preprint arXiv:1802.09841*, 2018.
[44] J. Kossen, S. Farquhar, Y. Gal, T. Rainforth, Active surrogate estimators: an active learning approach to label-efficient model evaluation, Adv. Neural Inf. Process. Syst. 35 (2022) 24557–24570.
[45] Y. Gal, R. Islam, Z. Ghahramani, Deep Bayesian active learning with image data, in: Proceedings of Machine Learning Research, 2017, pp. 1183–1192.
[46] M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, Annal. Math. Stat. 11 (1) (1940) 86–92.
[47] O.J. Dunn, Multiple comparisons among means, J. Am. Stat. Assoc. 56 (293) (1961) 52–64.

Tianxiang Wang received a bachelor's Degree in College of Computer Science and Technology, Huanghe Science and Technology University, Zhengzhou, China. Then He entered Henan Normal University, Xinxiang, China, in the same year as a graduate student. He completed his M.S. degree in computer science at Henan Normal University in July 2022, and then he became a Ph.D candidate in Nanjing University of Aeronautics and Astronautics. His-research interests include pattern recognition, medical image processing and deep learning.

Qun Dai completed her M.S. degree in Computer Science at Nanjing University of Aeronautics and Astronautics (NUAA), and then worked at the College of Information Science and Technology of NUAA as an assistant lecturer. There she received a Ph.D. degree in Computer Science in 2009. In 2010, she became an Associate Professor for the College of Computer Science and Technology of NUAA. Since 2015, she has become a Professor for the College of Computer Science and Technology of NUAA. Her research interests focus on neural computing, pattern recognition and machine learning.