Computers in Biology and Medicine 153 (2023) 106521

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

BrainTGL: A dynamic graph representation learning model for brain network analysis

Lingwen Liu [a], Guangqi Wen [a], Peng Cao [a,b,*], Tianshun Hong [a], Jinzhu Yang [a], Xizhe Zhang [c], Osmar R. Zaiane [d]

[a] Computer Science and Engineering, Northeastern University, Shenyang, China
[b] Key Laboratory of Intelligent Computing in Medical Image of Ministry of Education, Northeastern University, Shenyang, China
[c] School of Biomedical Engineering and Informatics, Nanjing Medical University, Nanjing, China
[d] Amii, University of Alberta, Edmonton, Alberta, Canada

ARTICLE INFO

Keywords:
Dynamic brain network
Graph classification
Resting-state fMRI
Spatio-temporal modeling

ABSTRACT

Modeling the dynamics characteristics in functional brain networks (FBNs) is important for understanding the functional mechanism of the human brain. However, the current works do not fully consider the potential complex spatial and temporal correlations in human brain. To solve this problem, we propose a temporal graph representation learning framework for brain networks (BrainTGL). The framework involves a temporal graph pooling for eliminating the noisy edges as well as data inconsistency, and a dual temporal graph learning for capturing the spatio-temporal features of the temporal graphs. The proposed method has been evaluated in both tasks of brain disease (ASD, MDD and BD) diagnosis/gender classification (classification task) and subtype identification (clustering task) on the four datasets: Human Connectome Project (HCP), Autism Brain Imaging Data Exchange (ABIDE), NMU-MDD and NMU-BD. A large improvement is achieved for the ASD diagnosis. Specifically, our model outperforms the GroupINN and ST-GCN by an average increase of 4.2% and 8.6% on accuracy, respectively, demonstrating its advantages in comparison to the state-of-the-art methods based on functional connectivity features or learned spatio-temporal features. The results demonstrate that learning the spatial–temporal brain network representation for modeling dynamics characteristics in FBNs can improve the model's performance on both disease diagnosis and subtype identification tasks for multiple disorders. Apart from performance, the improvements of computational efficiency and convergence speed reduce training costs.

1. Introduction

The brain is an exceptionally complex system and understanding its functional organization is the goal of modern neuroscience. By measuring the time-varying changes of blood-oxygen-level-dependent (BOLD) signal at rest, resting-state functional magnetic resonance imaging (rs-fMRI) becomes a powerful approach developed to characterize the brain activities and explore the intrinsic functional organization in the research of cognitive neuroscience, medical and clinical applications [1,2].

In our study, we focus on modeling rs-fMRI data as a brain network, where the nodes denote the brain regions and the edges represent the inter-regional functional associations. It is a graph classification task, which aims to predict the label of a given brain network. The brain networks can be analyzed with various graph learning-based methods. Motivated by the success of deep learning on grid data, graph convolutional networks (GCNs) [3] have been proposed to generalize the convolution operation on arbitrary graphs to address graph structure data. Recently, GCNs have achieved state-of-the-art results for learning the latent brain network representation for the classification in different neuro-disorders [4,5]. In this work, we focus on the analysis of brain networks with the GCNs methods.

Most of the related works construct a static brain network for analysis by estimating the correlations from BOLD signals, but they did not consider the spatio-temporal intertwining pattern in the brain networks [2,6,7]. Recent studies have shown that dynamic functional connection (FC) analysis provides valuable information for understanding the underlying functional brain activities [8,9]. Two recent works [10,11] consider both the spatial and temporal characteristics in the brain network for capturing spatio-temporal information. More specifically, a spatial–temporal graph convolutional network (ST-GCN) is proposed for learning the dynamic brain networks [10]. In [11], an end-to-end deep neural network combining temporal convolutional

* Corresponding author at: Computer Science and Engineering, Northeastern University, Shenyang, China.
E-mail address: caopeng@cse.neu.edu.cn (P. Cao).

https://doi.org/10.1016/j.compbiomed.2022.106521
Received 1 September 2022; Received in revised form 8 December 2022; Accepted 31 December 2022
Available online 6 January 2023

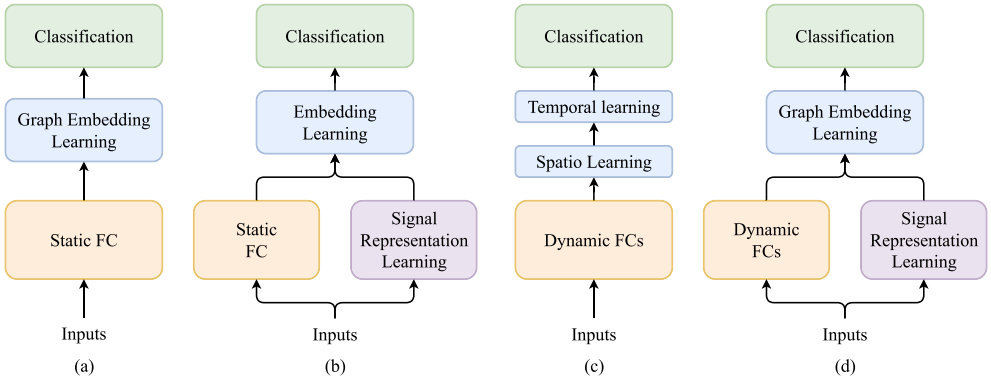


Fig. 1. Illustration of different brain network analysis methods with the input of BOLD signals in the rs-fMRI. (a) Most of the previous methods typically use Pearson's correlation based functional connectivity (FC) to characterize the temporal relationships between different brain regions during resting states, under an implicit assumption that FC of the human brain is stationary throughout the whole fMRI recording period. (b) The method usually assumes that the functional structure of brain network is constant (i.e., temporally stationary) throughout the recording period in rs-fMRI. It only concentrates on the one-dimensional BOLD series with TCN or LSTM models, but fails to consider the dynamic spatial dependency. (c) It learns the features from each functional connectivity network constructed by a segment of BOLD series into a high-level embedding independently, and then model the dynamic embeddings into a final representation. Although it models the dynamic property of functional connectivity networks, the major limitation is that the two-stage procedure is based on a clear separation of spatial and temporal feature learning stages, neglecting the temporal interaction of the spatial regions. (d) Our BrainTGL is designed to capture both the temporal dependency patterns from both the rs-fMRI time series and the dynamic graph structure by sufficiently taking advantage of sequential information conveyed in dynamic FC networks that could be informative to improve the diagnosis performance.

networks (TCNs) and GCNs is proposed to learn both the spatial and temporal components in rs-fMRI. Although they consider modeling the representation of the dynamic BOLD signals in the brain network [10,11], the graph structure in the network is still static since they ignore the dynamic variation with respect to the structure information in the brain network. These methods cannot capture the dynamic spatial relationships between the brain regions, which are crucial for understanding human actions. When modeling the dynamic brain network, both the BOLD signals in each brain region and the global topological correlation among the brain regions are very important for brain network analysis. However, most existing deep learning methods working on the rs-fMRI analysis fail to consider both aspects simultaneously.

Capturing the temporal dynamics of the brain network is challenging, because it is difficult to simultaneously model temporal dependency patterns from BOLD series and temporal structure changes of graph structure over time. To solve those challenges, we formulate the FBNs as dynamic graphs, and propose a dynamic graph embedding learning framework for brain networks analysis, named BrainTGL, by exploiting temporal graph information. More specifically, we first construct dynamic brain network series. Then, we propose an attention based graph pooling method and incorporate it into GCN. The pooling is able to preserve the critical graph structure by enhancing the important edges and removing the noisy edges. During the training of GCN, the node embedding is updated by aggregating information from the correlated neighbors according to the graph structure. Therefore, a clean and accurate graph structure is obtained to benefit for the GCN in learning node embeddings. Meanwhile, the pooling allows to determine different weights to edges within a graph, hence the final graph-level representation is informative for the GCN classifier. There exists a co-occurrence relationship between spatial and temporal domains. Moreover, we propose a dual temporal graph learning (DTGL) module to simultaneously model the dynamic information by capturing the spatio-temporal representation from two aspects in the coarsened and dynamic graphs. Furthermore, a multi-skip combination is introduced to extend the temporal span and hence ease the optimization process.

In addition to the graph classification task, graph clustering aims to explore the inherent subtypes of the same disorder without any supervision class label. The heterogeneity of brain disorders indicates that subjects may belong to different subtypes of the same disorder, which produces diverse graph structures and data distribution. However, subtypes of psychiatric disorders are primarily distinguished by clinical symptoms, ignoring the essential features of the disorder [12–14]. It is essential to develop an unsupervised clustering analysis framework for the discovery of patient disease subtypes, which has the potential to transform personalized medicine. In particular, with the BrainTGL for temporal graph embedding learning, we design a pseudo-label-based framework with a clustering-based label generation scheme.

We comprehensively evaluated BrainTGL on four real clinical applications covering: gender classification on the HCP dataset [15], ASD diagnosis on the ABIDE dataset [16], the diagnosis and subtype analysis of MDD and BD on the center NMU dataset. The results demonstrate the advantages of the proposed framework compared to existing state-of-the-art brain network analysis methods.

With BrainTGL, our contribution is threefold:

1. Capturing the spatio-temporal dynamics in functional brain networks is important for neurological disorders diagnosis and biomarkers exploitation. To solve it, we propose a dynamic brain network embedding learning framework by combining the advantages of GCN and LSTM to model the dynamic associations with an end-to-end scheme, which effectively models the co-occurrence relationships between brain regions across the dynamic time segments.
2. How to extract the critical graph structure in brain networks is still a challenging problem. To deal with it, we propose a graph structure learning with attention based graph pooling to remove the irrelevant FCs from the group level, which boosts the performance of the following temporal graph classification procedures.
3. Comprehensive evaluations on multiple datasets demonstrate that our method consistently outperforms state-of-the-art models for both the graph classification (supervised learning) on the tasks of gender classification, ASD diagnosis, MDD diagnosis and BD diagnosis, and graph clustering (unsupervised learning) on the tasks of MDD and BD subtype identification.

The rest of the paper is organized as follows. We discuss the related work on brain disease diagnosis in Section 2. A detailed mathematical formulation and framework description of BrainTGL and its variant clustering method (BrainTGL-C) are provided in Section 3. In Section 4, we evaluate the performance of the BrainTGL and BrainTGL-C by extensive experiments on the ABIDE, HCP and Center NMU datasets. Finally, we present the limitations and future works in Section 5, and conclude this work in Section 6.

2. Related work

Many efforts have been devoted to the automated brain diagnosis based on rs-fMRI. Current work is mainly divided into two directions: static brain network analysis and dynamic brain network analysis.

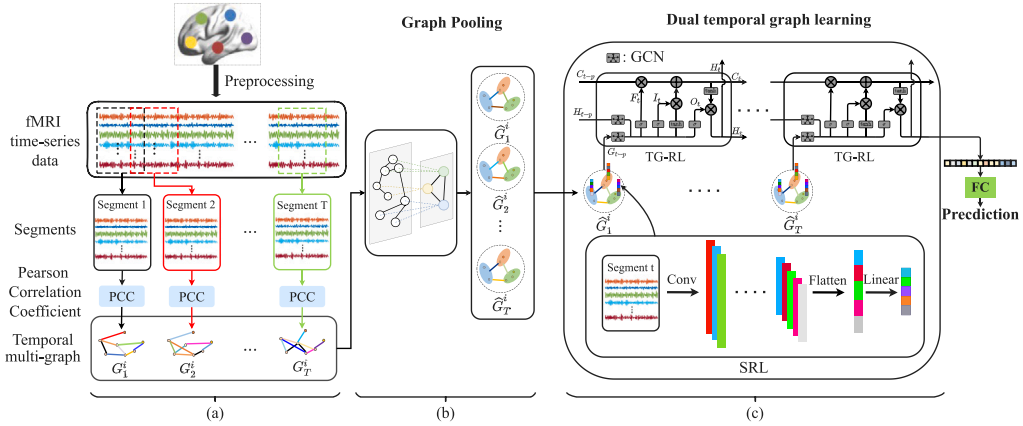


Fig. 2. The overall architecture of our proposed BrainTGL. (a): The construction of the dynamic graph series. (b): An attention based graph pooling is proposed to achieve temporal coarsened graph series. (c): A dual temporal graph learning is developed to sufficiently capture the temporal characteristics of the graph series from the BOLD signals level and graph level, respectively.

2.1. The static brain network analysis methods

Previous studies have successfully applied machine learning methods to identify cognitive impairment by constructing the static functional connectivity networks based on pair-wise temporal correlation between brain regions, as illustrated in Fig. 1(a)(b). For example, Jie et al. [6] introduced the graph-kernel based method that identified patients from normal controls by designing a measure to compute the topological similarity between the FCs. The major issue is that the feature extraction and classification are independent, which hinders the final performance. Many works [17–19] establish the CNN-based model to extract hierarchical topological features of brain networks for brain disease identification. Although these works overcome the limitations of feature extraction of traditional machine learning methods, the structure information between brain regions was ignored and they cannot capture the high-level topological representations. The structure information has been proven to be important for brain network learning [20].

In recent years, there has been an increased interest in graph convolution networks (GCNs). GCNs combine the advantages of both graph theory and deep learning approaches and have the potential to learn spatial representations in non-Euclidean domains. Recently, several studies have introduced GCN into the field of fMRI analysis, and have demonstrated the effectiveness of GCN-based models for brain disease classification. For example, Eslami et al. [7] proposed a joint learning method, which consists of an autoencoder for the reconstruction task and a single layer perceptron for ASD classification. Li [21] proposed an ensemble framework with hierarchical graph convolution network, which can capture intrinsic correlations among subjects to improve graph embedding learning for disease diagnosis. Although these methods have shown a potential advantage in improving the diagnosis performance for brain disease, they ignore capturing the temporal representations in rs-fMRI.

2.2. The dynamic brain network analysis methods

Recently, many works [8,9] suggest that the rich temporal representations in rs-fMRI benefits the embedding learning for disease classification. The dynamic FC can reliably monitor the changes of macroscopic neural activities underlying cognitive and behavioral decline. In an early stage, traditional machine learning methods have been widely used in the dynamic FC modeling and have shown significant success. For example, Monti et al. [22] proposed a smooth, incremental graphical lasso estimation (SINGLE) algorithm that regards fMRI time-series data as input to infer dynamic brain networks for each subject. Jie et al. [23] developed a manifold regularized multi-task feature learning framework to extract discriminative features from the dynamic FCs, and employed a multi-kernel SVM to diagnose Alzheimer's disease.

However, the traditional machine learning methods cannot sufficiently capture the potential information. Hence, deep learning models have recently emerged in the brain network analysis, and demonstrated the superior performance in the modeling dynamic association. Gadgil et al. [10] developed a spatial–temporal graph convolutional network (ST-GCN) by combining the functional connectivity and the temporal representations in BOLD signals leveraging the 1D convolutional kernels. Azevedo et al. [11] proposed a spatial–temporal learning module consisting of a GNN for spatial embedding learning and a TCN for temporal embedding learning. Different from ST-GCN, the input is the whole BOLD signals and the TCN is employed to capture the signal features. These two works focus on the dynamics of signal-level temporal signal, however, the structure of the brain network is still considered to be stationary, which implies that the variation of connections between brain regions is ignored. To solve it, Wang et al. [24] proposed an end-to-end temporal dynamics learning (TDL) method for the diagnosis of brain disease based on dynamic FCs. They first transformed rs-fMRI time series into dynamic FCs using overlapping sliding windows, then introduced a group-fused Lasso regularizer to capture the global temporal dynamics of these networks. However, the major limitation of TDL is the linear relationship for modeling dynamic brain networks. Modeling the correlation among brain regions or the correlation among temporal graph structures with nonlinear functions is able to provide enhanced flexibility and the potential ability to better capture the complex relationship. For example, Lin et al. [25] proposed a convolutional recurrent neural network (CRNN) which uses convolution to construct brain network and then extract sequential features via LSTM. Yin et al. [26] developed a dynamic graph representation learning framework based on GNN and LSTM to model the non-linear interrelationship among related nodes. The contrast between our model and theirs is that they learn features based on a clear separation of spatial and temporal feature learning stages, neglecting the temporal interaction of the spatial regions, while ours learns spatial and temporal features simultaneously.

Although the most dynamic brain network analysis methods (Fig. 1(c)) are able to model both the spatial and temporal features, there are still some drawbacks: (1) All of them only consider either signal-level temporal dynamics or dynamic FC-level temporal dynamics, failing to model both of them simultaneously; (2) They ignore the issues of the irrelevant correlations in each brain network and the inconsistency across brain networks; (3) Few work combines the network-based feature learning and the training of the classifier into a unified framework.

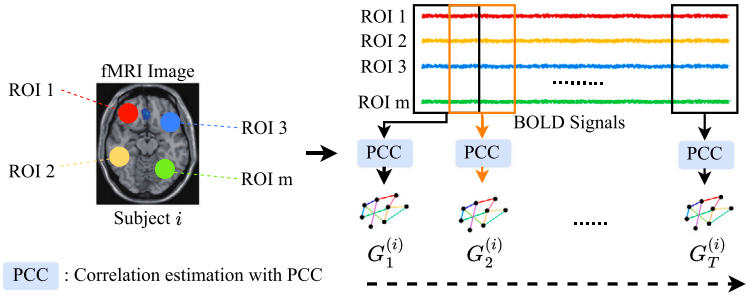
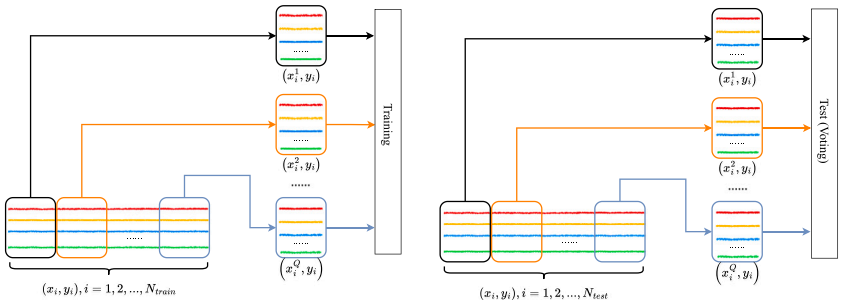


Fig. 3. The illustration of dynamic brain network construction from rs-fMRI images of the ith subject. To construct the dynamic brain networks, we divide the whole BOLD signals into small sequences via the sliding window with length L and stride S, and finally generate the temporal graphs series $G_1^{(i)}, G_2^{(i)}, \ldots, G_T^{(i)}$ of the ith subject.



(a) The augmentation for training (b) The augmentation for test

Fig. 4. Illustration of data augmentation for training and testing. (a): For each subject for training, it is divided into Q sub-sequences without overlapping. Each sub-sequence is labeled with the same class label with the subject. (b): For each unseen subject, the prediction result is voted by the results of the multiple sub-sequences.

3. Method

This section first introduces the problem statement of our tasks in Section 3.1 and then presents the proposed models in detail. In Section 3.2, the construction of dynamic brain networks is introduced. Section 3.3 shows the data augmentation method we used. Then in Sections 3.4 and 3.5, we present the modules of the attention based graph pooling and the dual temporal graph learning (DTGL). We discuss an ensemble strategy in Section 3.6. In addition, the variant method for clustering, named BrainTGL-C, is introduced in Section 3.7. Finally, we conclude the theoretical contributions in Section 3.8.

3.1. Problem statement

Formally, given the rs-fMRI time-series data $X = \{x_1^{(i)}, x_2^{(i)}, \ldots, x_n^{(i)}\}_{i=1}^N$ and their corresponding labels $Y = \{y_1, y_2, \ldots, y_N\}$. Note that $x_j^{(i)} \in \mathbb{R}^m$ represent the BOLD signal series of the jth ROI, $j \in \{1, 2, \ldots, n\}$, $y_i \in \{-1, 1\}$, m is the variable dimension, N is the amount of training samples and n is the amount of brain regions. We introduce a graph sequence to model the spatial–temporal graph structure for each subject. Let $\mathcal{D} = \{G_t^{(1)}, G_t^{(2)}, \ldots, G_t^{(N)}\}_{t=1}^T$ denote a dataset with each sample representing as a dynamic brain network series, where T denotes the number of time-series segments in rs-fMRI. Let $G_t^{(i)}$ denote the graph at the tth segment of the ith subject. Given a graph $G_t = \{V, X_t, A_t\}$, $A_t \in \mathbb{R}^{n \times n}$ denotes the corresponding adjacency matrix of G_t, $V = \{v_1, v_2, \ldots, v_n\}$ represents the set of nodes in G_t and X_t denotes the one-dimensional signals for the node set. In this work, we choose the CC-200 atlas, which divides the brain into 200 functionally homogeneous regions. The adjacency matrix A_t describes the graph structure of the brain network by calculating the correlation between each pair of brain regions with Pearson correlation coefficient (PCC). We formulate our task as a graph classification task and learn a mapping function $f : \{G_t\}_{t=1}^T \to Y$ (see Fig. 2).

3.2. Dynamic brain network series construction

In our work, we assume that the temporal information in fMRI time-series data benefits the disease diagnosis. As Fig. 3 shows, different from the static brain network methods which construct brain network by calculating the Person correlation with the whole BOLD signals, we design a sliding window technique, which divides the BOLD signal of each subject into many small segments of BOLD signal with a length of L and a stride of S, in order to sufficiently leverage the temporal information from the BOLD signal. The next step is to construct the adjacency matrix of the brain based on each small BOLD signal segment. For each small BOLD signal segment, the Person correlation is calculated to generate individual adjacency matrix. Finally, we obtain the dynamic brain network sequences for the training set with N subjects: $\mathcal{D} = \{G_t^{(1)}, G_t^{(2)}, \ldots, G_t^{(N)}\}_{t=1}^T$.

3.3. Data augmentation

Training deep learning models often requires a large number of samples to prevent overfitting. To alleviate the issue of insufficient data, we adopt a data augmentation scheme before constructing the dynamic brain network series. At first, we crop the input time courses to a fixed sequence length for all subjects, since the time courses have different lengths depending on the site. Then, we divide the whole sequence of each subject into several short sub-sequences of BOLD time series without overlap. The labels of the sub-sequences that are the same as its subject. During the training stage, we feed all the sub-sequences into the proposed BrainTGL model. At the testing stage, we apply the trained BrainTGL on each sub-sequence of the unseen data, and combine all the prediction scores by a voting scheme to produce a final subject-level prediction. The whole data augmentation of training data and test data is shown in Fig. 4.

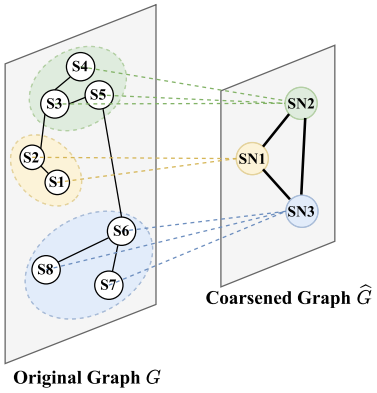


Fig. 5. The illustration of graph pooling for the coarsened graph generation.

3.4. Attention based graph pooling

When fMRI time-series data is converted into the dynamic brain network, it faces two problems: a large number of noisy connections and inconsistencies among the multi-site subjects. Graph pooling is a critical operation to coarsen an original graph with high level noisy edges for graph representation learning in GCN. The graph pooling procedure aims to preserve the global topology structure in brain networks and remove the noisy edges. In our work, we propose a temporal graph pooling with attention mechanism where weights are determined by GCN. Attention has been widely used in recent deep learning researches. We formulate graph pooling as a cluster assignment problem with attention mechanism. By hiding the non-indicative edges and highlighting the indicative edges, the nodes of an original graph are grouped into clusters. Formally, an original graph G is transformed into a coarsened graph $\hat{G} = \{\hat{V}, \hat{A}\}$ with a coarsened structure, where $\hat{V}$ denotes the set of supernodes in $\hat{G}$, $\hat{A} \in \mathbb{R}^{c\times c}$ denotes the corresponding adjacency matrix of $\hat{G}$, n denotes the number of nodes in G, and c denotes the number of supernodes in $\hat{G}$. To achieve it, we introduce a learnable parameter $F \in \mathbb{R}^{n\times c}$, which is optimized by our model for learning the important scores of the nodes for each supernode. F is formally defined as:

$$F_{ij} = \begin{cases} s_i, & i \in SN_j \\ 0, & i \notin SN_j \end{cases} \tag{1}$$

where s_i denotes the importance score of each node v_i in graph G. With the optimized F, we obtain a set of clusters as supernodes $\hat{V} = \{SN_1, SN_2, \ldots, SN_c\}$ and the weighted adjacency matrix of the supergraph $\hat{A} = F^T A F \in \mathbb{R}^{c\times c}$. Then the nodes of a cluster are pooled as one supernode to produce a coarsened graph as Fig. 5 shown. The superedge between supernodes is the aggregation of edges multiplying the node importance. Each superedge e_{ij} in $\hat{A}$ is defined as $s_i * w_{ij} * s_j$, where w_{ij} denotes the weight of edge e_{ij} in A.

Therefore, the learned edge weights help us identify the indicative edges. It is able to construct a coarsened and clean graph structure with only important edges by highlighting the critical connections and removing the irrelevant connections. Meanwhile, F is shared among all the graphs (brain networks) and achieved through learning from a group level. In summary, the advantage of our attention based graph pooling method is to group the nodes of the whole graph into some clusters by hiding the non-indicative connections and highlighting the task-relevant connections. It allows the model to focus more on task-relevant nodes of graphs for improving the downstream tasks, e.g. classification or clustering. Finally, the noisy connections are removed and the consistent coarsened temporal graphs are generated.

3.5. Dual temporal graph learning

To sufficiently model the spatio-temporal patterns of the brain activity, we propose a spatio-temporal modeling method called dual temporal graph learning (DTGL) that consists of S-RL module and TG-RL module to fully learn temporal characteristics in fMRI data from two aspects: one-dimensional BOLD signals and multivariable temporal graphs, respectively. It is quite different from the previous work.

Firstly, to learn the temporal embedding from the one-dimensional BOLD signal, we propose a signal representation learning (S-RL) module via a stack of convolutional layers. The temporal embedding of node is learned after l steps of convolutional layers as follows:

$$e^{(l+1)}(u) = \sum_{s=0}^{U-1} e^{(l)}(u-s) * \mathcal{K}^{(l)}(s) \tag{2}$$

where $\mathcal{K}^{(l)}$ is a convolutional kernel of lth layer with a kernel size of U and u denotes the elements in BOLD signal. The initial embedding of each supernode is obtained with a simple max pooling on the temporal embedding of nodes within the corresponding supernode.

Long short-term memory (LSTM) is a feedback deep learning architecture proposed to solve the poor learning ability of RNNs in longer sequences because of the vanishing gradient problem. A standard LSTM unit consists of a cell (C_t), an input gate (I_t), an output gate (O_t) and a forget gate (F_t). However, there are two problems for the traditional LSTM:

(1) the fully connected operator within LSTM ignores the spatial correlation;

(2) the fixed skip length within LSTM is constrained due to its inability to take advantage of the dependencies with variable lengths.

To solve these limitations, we design a temporal graph representation learning (TG-RL) module to model temporal characteristics at the graph level. To this end, we combine the graph convolution with the LSTM model to model the spatio-temporal representation for the dynamic brain networks effectively. The TG-RL module also has three gates: the input gate, the forget gate, and the output gate. This module is fed with the temporal coarsened graph series, which contain the initial node embedding obtained by Eq. (2). In the module, the LSTM component is adopted to learn the temporal embedding of all the dynamic coarsened graphs within a dynamic network series. For each coarsened graph, the GCN component is applied to capture the graph structural properties of nodes as well as the relationship between them. Hence, the operation of each gate is a stack of graph convolutional layers to exploit the spatial relationships between brain regions according to the graph structures learned by the graph pooling. Furthermore, we propose a multi-skip scheme in the TG-RL module to solve them via a multi-skip combination to capture multi-level temporal information for long and short hops. With the multi-skip scheme, the TG-RL cell contains three inputs: H_{t-p}, $\hat{G}_t$ and $E_t^{\mathcal{G}}$, where H_{t-p} denotes the hidden state obtained in $(t-p)$-th step, p is the number of hidden cells skipped through, $\hat{G}_t$ is the tth graph in the dynamic graph series and $E_t^{\mathcal{G}}$ is the corresponding features of the tth graph in the input graph series. The input, hidden state and cell memory of TG-RL are all graph-structure rather than vectors in the traditional LSTM. The updating process is defined as

$$\textbf{InputGate}: \quad I_t = \textbf{sigmoid}(W_i * \textbf{Gconv}(\hat{G}_t) + \hat{W}_i * \textbf{Gconv}(H_{t-p})) + b_i \tag{3}$$

$$\textbf{ForgetGate}: \quad F_t = \textbf{sigmoid}(W_f * \textbf{Gconv}(\hat{G}_t) + \hat{W}_f * \textbf{Gconv}(H_{t-p})) + b_f \tag{4}$$

$$\textbf{OutputGate}: \quad O_t = \textbf{sigmoid}(W_o * \textbf{Gconv}(\hat{G}_t) + \hat{W}_o * \textbf{Gconv}(H_{t-p})) + b_o \tag{5}$$

$$\textbf{InputModulation}: \quad U_t = \textbf{relu}(W_c * \textbf{Gconv}(\hat{G}_t) + \hat{W}_c * \textbf{Gconv}(H_{t-p})) + b_c \tag{6}$$

$$\textbf{CellMemory}: \quad C_t = \textbf{Tanh}(I_t * U_t + F_t * C_{t-p}) \tag{7}$$

$$\textbf{Output}: \quad H_t = O_t * \textbf{Tanh}(C_t) \tag{8}$$

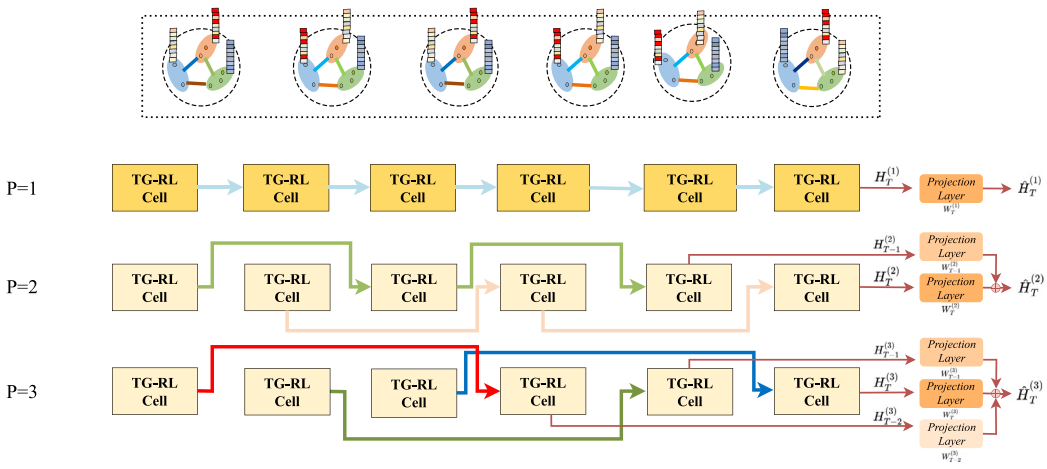


Fig. 6. The illustration of multiple skip connection with variable lengths. P indicates the skip length of TG-RL.

where I_t, F_t, O_t, U_t, H_t and C_t are input gate, forget gate and output gate like LSTM, modulated input, the hidden state and cell memory, respectively. All of them are also graph-structured data. W_i, W_f, W_o, W_c, $\hat{W}_i$, $\hat{W}_f$, $\hat{W}_o$ and $\hat{W}_c$ are learned weights of different gates. b_i, b_f, b_o and b_c are the bias vector parameters of gates respectively.

With the node embedding matrix $E_t^{\mathcal{G}}$ and the corresponding adjacency matrix $\hat{A}_t$ of $\hat{G}_t$, the graph convolution $\mathbf{Gconv}(\cdot)$ in lth layer can be defined as:

$$E_t^{\mathcal{G}^{(l+1)}} = \mathbf{Gconv}(\hat{G}_t) = \mathbf{relu}(\hat{A}_t E_t^{\mathcal{G}^{(l)}} W_G^{(l)}) \tag{9}$$

where $W_G^{(l)}$ is a learnable graph convolution weight matrix of lth layer, $E_t^{\mathcal{G}^{(l+1)}}$ is the node embedding matrix computed after l steps of graph convolution. Noting that $E_t^{\mathcal{G}^{(0)}}$ is the graph initial embedding of $\hat{G}_t$ which is equal to E_t, each item of which is obtained by Eq. (2).

When we choose multiple skip lengths ($p = 1, 2, \ldots, P$), our TG-RL is able to capture long-term dependencies with variable lengths in the temporal graph series. From Fig. 6, it can be seen that:

- When $P = 1$ (the skip length is 1), thus the final output is $\hat{H}_T^{(1)} = W_T^{(1)} H_T^{(1)}$;
- When $P = 2$ (the skip length is 2), thus the final output is $\hat{H}_T^{(2)} = W_T^{(2)} H_T^{(2)} + W_{T-1}^{(2)} H_{T-1}^{(2)}$;
- When $P = 3$ (the skip length is 3), thus the final output is $\hat{H}_T^{(3)} = W_T^{(3)} H_T^{(3)} + W_{T-1}^{(3)} H_{T-1}^{(3)} + W_{T-2}^{(3)} H_{T-2}^{(3)}$;

where $H_T^{(1)}$, $H_T^{(2)}$ and $H_T^{(3)}$ are the outputs of the Tth step with different skip lengths in our TG-RL.

Finally, we combine the multiple outputs of TG-RL with multiple skip connections with a dense layer to produce a final embedding as follows:

$$\hat{H}_T^C = \sum_{p=1}^{P} \sum_{i=T}^{T-p+1} W_i^{(p)} H_i^{(p)} + b\,, \tag{10}$$

where $H^{(1)}$, $H^{(2)}$, …, $H^{(P)}$ are the outputs with "multi-skip" connections and $W^{(p)}$ are learnable matrices.

3.6. Ensemble

The hyperparameter of window length influences the performance of our algorithm. It is time-consuming to find an optimal solution. To alleviate this, we adopt a multi-time window ensemble strategy. As shown in Fig. 7, we train multiple BrainTGLs on the temporal graph series with different window lengths. Finally, these trained models are combined to achieve a final prediction with a voting scheme.

3.7. Variation of braintgl for unsupervised clustering

Apart from the modeling of dynamic FC characteristic, subtype identification by rs-fMRI also draws increasing attention in the neuroscience community as a disease analysis. Unsupervised clustering has been proven to be useful in the analysis of subtypes and shows great promise for application to brain disease. We extend our BrainTGL to an unsupervised learning framework, named BrainTGL-C. Our BrainTGL-C incorporates the spatial–temporal graph embedding learning and cluster assignments learning into an end-to-end deep learning model. The clustering framework iteratively groups the samples with a hierarchical clustering algorithm, and uses the subsequent assignments as supervision labels to learn the parameters of the BrainTGL. Our network structure is shown in Fig. 8. The framework is mainly divided into three stages: (a) spatial–temporal graph embedding learning, (b) hierarchical clustering and (c) fine-tuning the data with pseudo-labels.

More specifically, we initialize all pseudo labels at the beginning of each iteration until the quality of pseudo labels stabilizes and the model performance no longer improves. Specifically, for each iteration, at the beginning of hierarchical clustering, we regard N instances as N different identities and initialize all pseudo labels. We calculate all pairwise distance between instances in the training dataset and generate a $N \times N$ distance matrix. According to distance matrix and clustering distance measurement, we will merge the nearest clusters in each step until the cluster number K is achieved. We regard samples in the same cluster to have the same pseudo labels, and generate pseudo labels to guide model training in the end. We introduce cross-entropy loss as the objective function of the clustering framework.

3.8. Theoretical contribution

Modeling spatio-temporal dynamics in functional brain networks is critical for underlying the functional mechanism of neurological disorders including ASD, MDD and BD. The current work faces two challenges involving: how to effectively learn the spatio-temporal data representation for training a predictive diagnostic model, and how to capture the critical graph structure for exploring interpretability in the brain diseases.

(a) Feature learning for spatio-temporal data

Spatio-temporal embedding learning is becoming growingly important in the big data era with the increasing availability in various domains including environment and climate (e.g. wind prediction and precipitation forecasting), public safety (e.g. crime prediction), intelligent transportation (e.g. traffic flow prediction), human mobility (e.g. human trajectory pattern mining). However, most sequential models just emphasize the dependencies among sequence nodes, ignoring

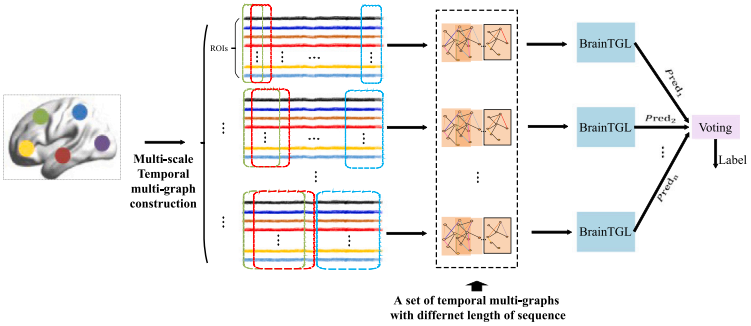


Fig. 7. An ensemble of BrainTGL with multiple window parameters for sliding windows.

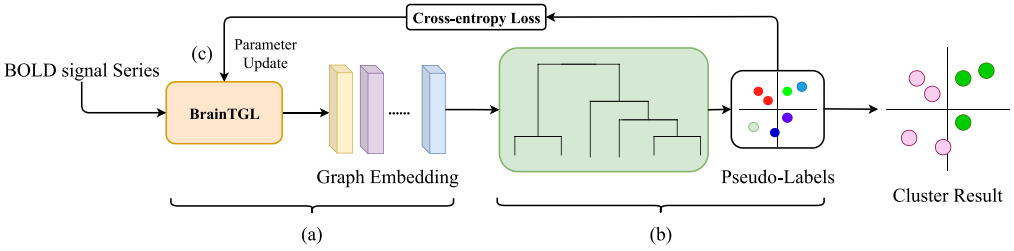


Fig. 8. The framework of BrainTGL-C. (a): Spatial–temporal graph embedding learning based on the BrainTGL. (b): Hierarchical graph clustering. (c): Fine-tuning training with pseudo-labels. It clusters the graph embeddings from BrainTGL and generates the pseudo-labels. Finally, the model is optimized for supervised learning by the pseudo-labels generated from the clustering.

other correlations like spatial and temporal relations among them. In the study of brain network analysis, there are complicated co-occurrence relationships between brain regions across the dynamic time segments. We argue that these spatial–temporal relationships are essential for learn the robust representation. In this study, we propose a unified model which is capable of handling spatio-temporal data for capturing the dynamic functional connectivity. Compared with LSTM, the inner operator of TG-RL is graph convolutional calculation. Due to the graph convolutional operation within our TG-RL, both the cell memory and hidden state are able to capture temporal dynamics and spatial structural information. Our model can not only capture discriminative features in spatial configuration and temporal dynamics, but also explore the co-occurrence relationship between spatial and temporal domains. Moreover, it is worth noting that our model is a general and effective framework for learning representation of temporal data with graph structure. Furthermore, our model is a more efficient network capable of modeling spatio-temporal data at a lower computational cost.

(b) Graph structure learning for the graph data with complex structure

Graph Convolutional Networks (GCNs) are widely used for analyzing graph-structured data because of their ability to exploit the rich information inherent in the graph structures and attributes. However, it is inevitable that the provided graph is incomplete and noisy, resulting in inaccurate predictions, which necessitates learning robust representations for real-world problems (e.g. node classification, recommendation, information retrieval and medical analysis). To alleviate the issue of complex structure in brain networks, we introduce graph structure learning into the GCN model to highlight the critical connections and remove the irrelevant connections via an attention pooling with a supervision scheme. The aim is to learn a critical graph structure and generate more robust and biologically meaningful functional connections. Another problem is that there always exists inter-site heterogeneity in the multi-site data, which is always ignored by the traditional GCN model. In the ABIDE dataset, there exists 17 sites, where the image contrast, resolution and noise levels between sites are different. The inter-site variation in the multi-center data hinders the direct application of the traditional brain network embedding learning methods. Training a single predictive model on a multi-center dataset is more challenging to capture the heterogeneous data, and this inconsistency of brain network structures limits the exploration of the effective biomarkers. Our attention based pooling allows the graph structure learning to be aware of the group level for solving the inter-site variation issue. The learned consistent graph structure by our proposed graph structure learning enables our framework to provide an interpretable framework. Moreover, our proposed graph structure learning is a general learning model for graph data, not limited to brain network.

4. Experiments

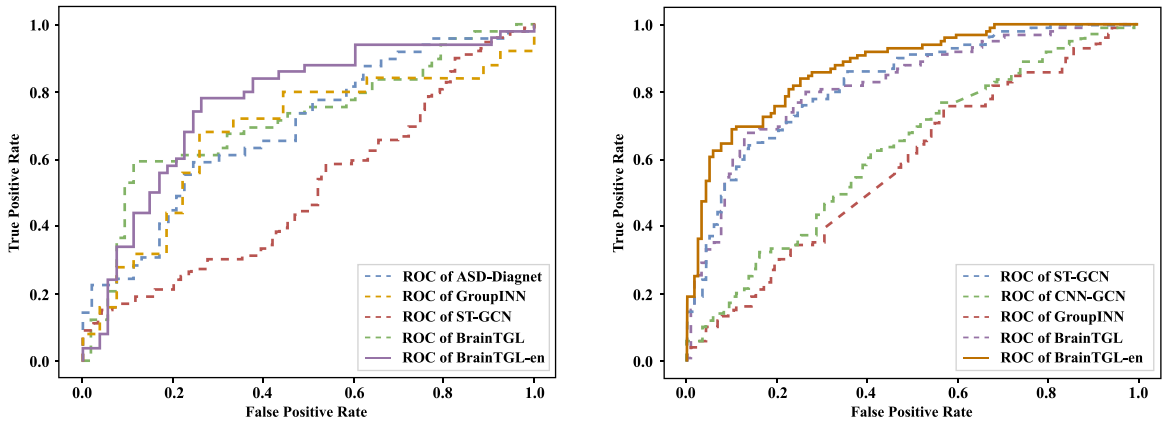
4.1. Datasets and environment

We evaluated our proposed framework on four challenging datasets for graph classification and clustering tasks: HCP (Human Connectome Projects dataset) and ABIDE (Autism Brain Imaging Data Exchange dataset), NMU-MDD (Major depressive disorder datasets collected by Nanjing Medical University) and NMU-BD (Bipolar disorder datasets collected by Nanjing Medical University).

ABIDE: The ABIDE database contains 1112 subjects, which are collected from 17 acquisition sites. After the preprocessing, we obtained 871 high-quality rs-fMRI, comprising 403 individuals with ASD aged 18.069±8.415 and 468 normal controls aged 17.321±7.343. We further selected 512 individuals whose sequence length of ROIs' corresponding BOLD time-series is between [176, 250].

HCP: Human Connectome Project (HCP) S1200 contained 1096 young adults with the rs-fMRI data. The first session for each subject is used and five rs-fMRI data with less than 1200 frames are excluded. It results in the dataset containing 498 females aged 29.559±3.6098 and 593 males aged 27.895±3.6103. The cortical surface was parcellated into 22 major brain regions.

Center NMU dataset: The center NMU (Nanjing Medical University) dataset is provided and permitted for use by Nanjing Medical University. In the process of data pre-processing, we deal with data by using dpabi and divide the whole brain into 90 brain regions based on



(a) ASD vs. NC (b) Male vs. Female

Fig. 9. The ROC of multiple computing methods.

Table 1
The information regarding the hardware environment in our experiments.

Category	Specifications
System	Ubuntu 20.04.3 LTS (GNU/Linux 5.13.0-30-generic x86_64)
CPU	2 × Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10 GHz 8-core
Memory	64 GB
GPU	4 × NVIDIA GeForce GTX 1080 8G

Automated Anatomical Labeling (AAL) for analysis. They included spatial normalization to Echo Planar Imaging (EPI) template of standard Montreal Neurological Institute (MNI) space (spatial resolution 3 mm × 3 mm × 3 mm), spatial and temporal smoothing with a 6 mm × 6 mm × 6 mm Gaussian kernel and filter processing by adopting 0.01–0.08 Hz low-frequency fluctuations to remove interference signals. The dataset included 246 health controls (152 females and 94 males, aged 26.89±6.14), 181 MDDs (138 females and 43 males, aged 16.97±5.01) and 146 BDs (103 females and 43 males, aged 17.24±4.03), who were scanned at a single site with identical inclusion and exclusion criteria. In our experiments, we divide the NMU dataset into NMU-MDD (246 HCs and 181 MDDs) and NMU-BD (246 HCs and 146 BDs) datasets.

We implement our model in PyTorch, and use a machine equipped with four GPUs to accelerate the model training. The details are shown in Table 1.

4.2. Comparison with the prior works on the brain network classification

To evaluate the effectiveness of the proposed BrainTGL and BrainTGL-en, which is the ensemble model of multiple BrainTGLs on the temporal graph series with different window lengths, on the graph classification on the dynamic brain network, we compared our proposed method with the current state-of-the-art approaches on ABIDE and HCP including: **SVM**, **LSTM**, **GCN**[4], **GroupINN** [2], **ASDDiagnet** [7], **ST-GCN** [10] and **CNN-GCN** [11]. For both the NMU-MDD and NMU-BD, we compared our BrainTGL model with the competing approaches including: **GroupINN**, **ST-GCN**, **ASDDiagnet** and **BrainnetCNN** [27].

In our experiments, we employed a 5-fold cross-validation to evaluate our methods. The results of the comparisons in terms of accuracy, AUC, sensitivity and specificity are reported in Tables 2 and 3. The area under the receiver operating characteristic (ROC) curves are further plotted in Fig. 9. We can see that our methods can consistently and substantially beat the previous brain network classification methods on both tasks. Our method BrainTGL-en obtained the accuracies of

Table 2
Performance comparison of various methods on HCP dataset and ABIDE dataset, where the colors of red and blue denote the top two best results, respectively.

Dataset	Method	AUC (%)	ACC (%)	SEN (%)	SPE (%)
HCP	SVM	62.2 ± 1.29	61.5 ± 1.26	64.1 ± 1.31	61.1 ± 1.20
	LSTM	64.1 ± 3.18	64.3 ± 2.99	62.9 ± 3.01	65.6 ± 2.87
	GCN[4]	64.6 ± 1.36	65.1 ± 1.93	62.4 ± 1.86	67.8 ± 1.89
	GroupINN [2]	68.8 ± 2.19	69.3 ± 2.32	63.0 ± 1.14	74.6 ± 3.90
	ST-GCN [10]	76.5 ± 2.21	76.3 ± 1.72	68.3 ± 1.91	**83.9 ± 1.77**
	CNN-GCN [11]	62.3 ± 2.64	61.1 ± 1.36	60.5 ± 2.92	68.2 ± 1.69
	BrainTGL	**79.9 ± 2.35**	**80.0 ± 2.3**	**79.4 ± 1.71**	80.6 ± 1.89
	BrainTGL-en	**81.1 ± 1.96**	**82.8 ± 1.61**	**79.8 ± 1.98**	**83.2 ± 1.74**
ABIDE	SVM	59.3 ± 1.30	61.1 ± 1.23	59.0 ± 1.22	58.7 ± 1.23
	LSTM	60.9 ± 3.22	61.4 ± 4.50	58.8 ± 3.89	63.5 ± 4.11
	GCN[4]	63.5 ± 1.22	62.7 ± 2.19	59.7 ± 2.04	61.3 ± 1.92
	GroupINN [2]	63.4 ± 2.40	63.4 ± 2.35	**63.6 ± 1.74**	63.2 ± 1.50
	ST-GCN [10]	59.0 ± 1.43	59.2 ± 2.12	54.9 ± 1.56	63.2 ± 1.91
	ASDDiagnet [7]	64.0 ± 2.16	**67.5 ± 2.3**	62.3 ± 1.31	67.4 ± 2.91
	BrainTGL	**65.3 ± 1.41**	65.4 ± 1.68	**62.6 ± 1.88**	**68.1 ± 1.79**
	BrainTGL-en	**67.6 ± 1.52**	**67.8 ± 1.66**	62.2 ± 1.92	**72.9 ± 1.86**

67.8%, 82.8%, 73.2%, 72.0% on ABIDE, HCP, NMU-MDD and NMU-BD, respectively. In addition, it can be seen that our model can achieve a comparable performance without the ensemble scheme with the accuracies of 65.4%, 80.0%, 69.8%, 68.0%, respectively. Specifically, compared with the traditional method SVM, BrainTGL achieves a better performance, indicating the effective capability of deep learning methods. Compared with the basic GCN or LSTM, a large margin of about 14.9% and 15.7% improvement on the HCP dataset. About 2.7% and 4.0% improvements have been achieved on the ABIDE dataset. The remarkable results imply the effectiveness of appropriately integrating GCN with LSTM for modeling the spatio-temporal data. GroupINN, which focuses on the spatial graph convolution, potentially neglects the sufficient temporal information hidden in the rs-fMRI. Compared with CNN-GCN and ST-GCN, which also focus on the spatio-temporal modeling, our proposed methods perform better on all the datasets. The main reasons are: (1) they ignore the complicated and inconsistent graph structure within the multi-center subjects, which poses a significant challenge to learn a good representation; (2) they neglect the graph-level temporal dynamics when capturing the signal-level temporal dynamics in the BOLD signal. In summary, compared to the competing methods, our proposed methods provide more discriminative and comprehensive information by jointly considering the potential relations of different brain regions in both spatial and temporal dimensions from the perspectives of the graph level and signal level.

Table 3
Performance comparison of various methods on NMU-MDD dataset and NMU-BD dataset, where the colors of red and blue denote the top two best results, respectively.

Dataset	Method	AUC (%)	ACC (%)	SEN (%)	SPE (%)
NMU MDD	GroupINN [2]	63.6 ± 1.54	64.5 ± 1.29	**69.4 ± 1.32**	64.8 ± 1.62
	ST-GCN [10]	53.1 ± 1.29	57.8 ± 2.32	51.2 ± 1.41	54.5 ± 1.66
	ASDDiagnet [7]	62.4 ± 1.57	66.8 ± 2.13	65.9 ± 2.46	69.3 ± 2.41
	BrainnetCNN [27]	65.8 ± 1.79	67.9 ± 2.27	67.5 ± 1.55	50.6 ± 2.42
	BrainTGL	**66.2 ± 2.26**	**69.8 ± 2.16**	68.1 ± 1.95	**70.6 ± 1.66**
	BrainTGL-en	**68.9 ± 1.14**	**73.2 ± 1.93**	**70.2 ± 1.47**	**78.0 ± 1.51**
NMU BD	GroupINN [2]	61.2 ± 1.51	65.9 ± 2.21	66.1 ± 1.63	67.1 ± 2.42
	ST-GCN [10]	56.1 ± 2.32	59.7 ± 2.64	53.5 ± 1.93	61.0 ± 1.87
	ASDDiagnet [7]	**70.1 ± 3.21**	**71.2 ± 2.33**	**67.5 ± 1.73**	68.3 ± 1.54
	BrainnetCNN [27]	64.5 ± 1.77	68.8 ± 2.28	64.4 ± 1.62	51.4 ± 1.75
	BrainTGL	60.1 ± 1.34	68.0 ± 1.63	65.9 ± 1.51	**68.8 ± 1.57**
	BrainTGL-en	**70.8 ± 1.70**	**72.0 ± 1.52**	**76.1 ± 1.62**	**73.9 ± 1.69**

Table 4
The ablation study on the ABIDE dataset. **Aug.** indicates the data augmentation, **P.** indicates the graph pooling and **En.** indicates the ensemble strategy.

Method	GCN	LSTM	**P.**	**Aug.**	DTGL	**En.**	ACC (%)	SEN (%)	AUC (%)	SPE (%)
GCN	√						56.0	60.6	57.2	59.1
GCN-LSTM	√	√					58.4	59.1	58.8	57.1
DTGL					√		61.1	63.3	58.0	63.8
DTGL-aug				√	√		62.3	64.1	61.2	62.2
DTGL-aug-en				√	√	√	64.5	65.9	63.6	65.1
P-DTGL-aug(BrainTGL)			√	√	√		65.4	**62.6**	65.3	68.1
P-DTGL-aug-en (BrainTGL-en)			√	√	√	√	**67.8**	62.2	**67.6**	**72.9**

Table 5
The ablation study on the HCP dataset. **Aug.** indicates the data augmentation, **P.** indicates the graph pooling and **En.** indicates the ensemble strategy.

Method	GCN	LSTM	**P.**	**Aug.**	DTGL	**En.**	ACC (%)	SEN (%)	AUC (%)	SPE (%)
GCN	√						59.0	62.2	58.6	55.1
GCN-LSTM	√	√					58.3	60.5	57.4	54.8
DTGL					√		71.3	71.3	71.0	70.8
DTGL-aug				√	√		73.7	64.4	81.8	81.4
DTGL-aug-en				√	√	√	75.4	67.9	80.1	81.7
P-DTGL-aug(BrainTGL)			√	√	√		80.0	79.4	79.9	80.6
P-DTGL-aug-en (BrainTGL-en)			√	√	√	√	**82.8**	**79.8**	**81.1**	**83.2**

4.3. Ablation study

To demonstrate the efficiency of our framework design, a careful ablation study was conducted. Specifically, the comparison was conducted between our method and the intermediate method or basic method with a single component or a combination of multiple components. The experimental results are reported in Tables 4 and 5. From the tables, it can be seen that P-DTGL-aug-en (BrainTGL-en) yields the best performance with respect to all the metrics. Moreover, we can see that DTGL achieves better results than GCN-LSTM, which demonstrates that the temporal relations and structural relations are complementary and leads to the conclusion that a simple LSTM module cannot capture well the temporal information in fMRI data. By comparing DTGL-aug and DTGL, it can be clearly observed that the insufficient number of samples in the dataset is one of the main factors hindering the performance of our model. The data augmentation facilitates the learning of the dynamic brain network and improves the discrimination capability of deep learning models. Additionally, we can see that the proposed P-DTGL-aug-en (BrainTGL-en) achieves better results than DTGL-aug-en, confirming that graph pooling plays a crucial role in modeling dynamic brain networks. At last, the results show that the ensemble strategy provides improved performance to eliminate its determination of the window size.

4.4. Discussion

4.4.1. The impact of attention graph pooling

We also apply *t*-SNE on the original brain network graphs and pooled brain network graphs to measure the distribution of graphs on the 2-dimensional space. The *t*-SNE results are shown in Fig. 10, with each point representing a graph sample and different colors denotes the graph samples from different acquisition sites. From Fig. 10(a), some obvious semantically inconsistent samples are evident before graph pooling. After graph pooling, as the distribution of Fig. 10(b) shows, the inconsistency samples is significantly decreased.

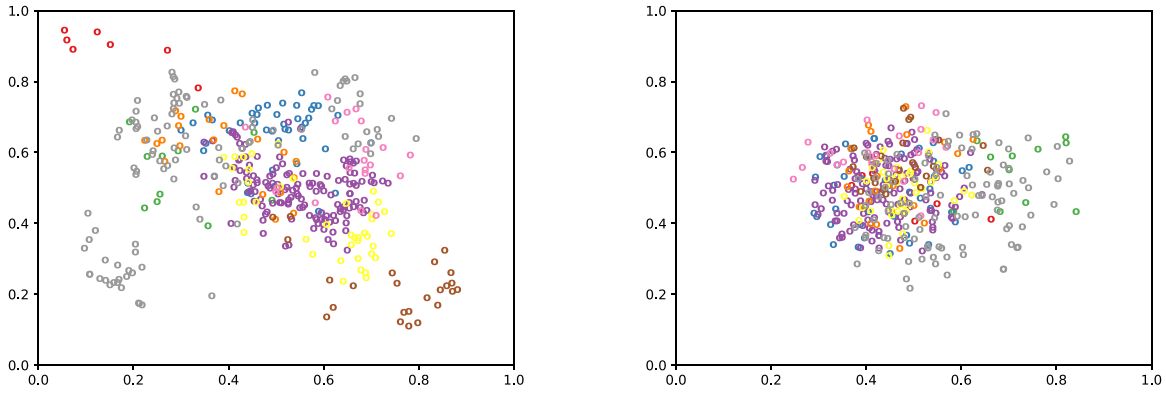
4.4.2. The impact of the supernode number

In this subsection, we aim to answer the questions: Does the attention based graph pooling solve the noisy connections in the brain network and the data inconsistency in the multi-site dataset? To show the comparison among the different supernode numbers of our model, we employ BrainTGL with the supernode numbers from 5 to 20. The ACC (%) performance is shown in Fig. 11. It can be found that the performance is improved with the number of supernodes increasing until 8 supernodes, then it tends to decline. The result suggests that more supernodes help capture critical graph structure better. Nevertheless, too many supernodes will inevitably introduce noisy nodes and edges to the model, leading to overfitting.

4.4.3. Comparison of pooling methods

There have also been attempts to design graph pooling methods for learning hierarchical features that are crucial for graph representation and classification.

DIFFPOOL [28]: DIFFPOOL is a graph pooling method that learns a cluster assignment matrix over the nodes using the output of a GNN model.



(a) The t-SNE result before graph pooling. (b) The t-SNE result after graph pooling.

Fig. 10. The embedded t-SNE representations of brain networks before and after the proposed graph pooling.

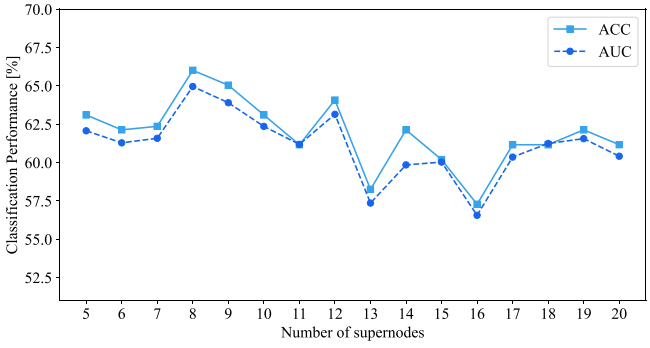


Fig. 11. The performance of the attention graph pooling with different supernode numbers in BrainTGL.

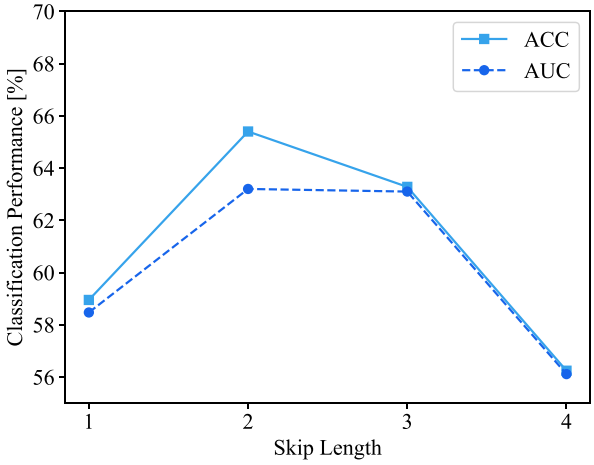


Fig. 12. Performance of TG-RL with different lengths of skips.

SAGPool [29]: SAGPool is a graph pooling module that introduces an attention mask process to capture the key nodes through a self-attention mechanism.

AttPOOL [30]: AttPool is a graph pooling module that selects significant nodes for graph representation adaptively based on an attention based mechanism.

The major differences among our pooling and the comparable pooling methods are:

(1) Our proposed attention based graph pooling method achieves information aggregation based on graph structural representation clustering, while the DIFFPOOL, SAGPool and AttPOOL methods are based on attention of node features.

(2) The DIFFPOOL, SAGPool and AttPOOL focus on the node selection in pooling while our method focuses on the aggregation of edges, which are more critical for the brain network analysis.

Specifically, we replace the attention based graph pooling module in our model with the pooling modules of DiffPOOL, SAGPool and ATTPOOL, and keep the backbones and other parts of the networks the same as our BrainTGL.

The results of the comparative graph pooling methods are shown in Table 6. From the comparison of different pooling methods, we can find that our graph pooling method which performs information aggregation based on graph clustering during the pooling operation, achieves better performance. It reveals that the information aggregation based on attention based graph clustering can be helpful for graph spatio-temporal representation learning.

4.4.4. The impact of different skip lengths in TG-RL

To explore the effects of different values of skip length, we chose four skip lengths: 1, 2, 3, 4 to train BrainTGL model on the ABIDE dataset. The result in Fig. 12 shows that the value of skip length has a significant impact on the classification performance, which demonstrates that it is an important factor for BrainTGL. Specifically, when the skip length value is small, the temporal information cannot be sufficiently captured. When the skip length value becomes larger, the temporal information is prone to be inevitably lost. We find that when the skip is 2, our model achieves the best performance, which shows the appropriate skip length is capable of helping the model capture appropriate temporal information.

Table 6
The comparison of the graph pooling methods on ABIDE dataset.

Method	ACC (%)	SEN (%)	SPE (%)	AUC (%)
DIFFPOOL [28]	61.2 ± 1.12	48.8 ± 1.20	**72.8 ± 1.95**	60.8 ± 1.96
SAGPool [29]	62.9 ± 1.70	54.0 ± 2.48	68.3 ± 2.22	63.4 ± 1.57
ATTPOOL [30]	61.6 ± 1.13	59.2 ± 1.84	64.7 ± 2.30	62.1 ± 2.55
BrainTGL	**65.3 ± 1.61**	**65.4 ± 1.41**	62.6 ± 1.88	**68.1 ± 1.79**

4.4.5. The model complexity

The computational complexity is an important measure for predictive models. In this study, we evaluate the comparable models in terms of the parameter number, FLOPs and training time on the ABIDE dataset. Notably, It can be observed that our model is more parameter-efficient compared with other models (ASDDiagnet, ST-GCN and GroupINN) as shown in Fig. 13. BrainTGL has only 0.19M parameters, requiring less trainable parameters. Specifically, BrainTGL

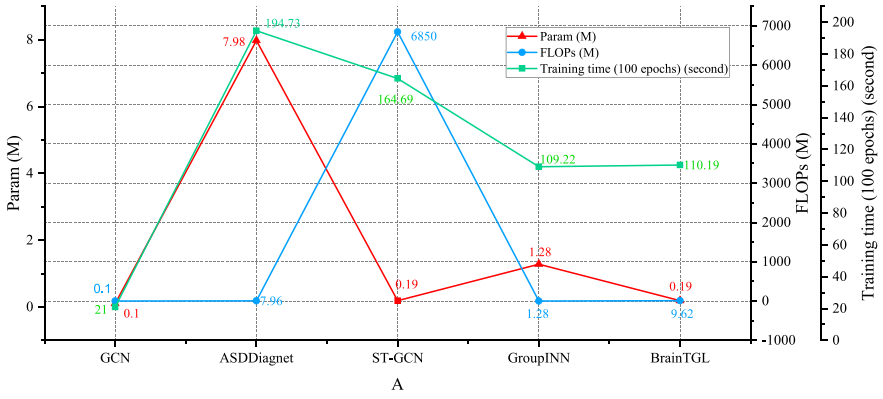
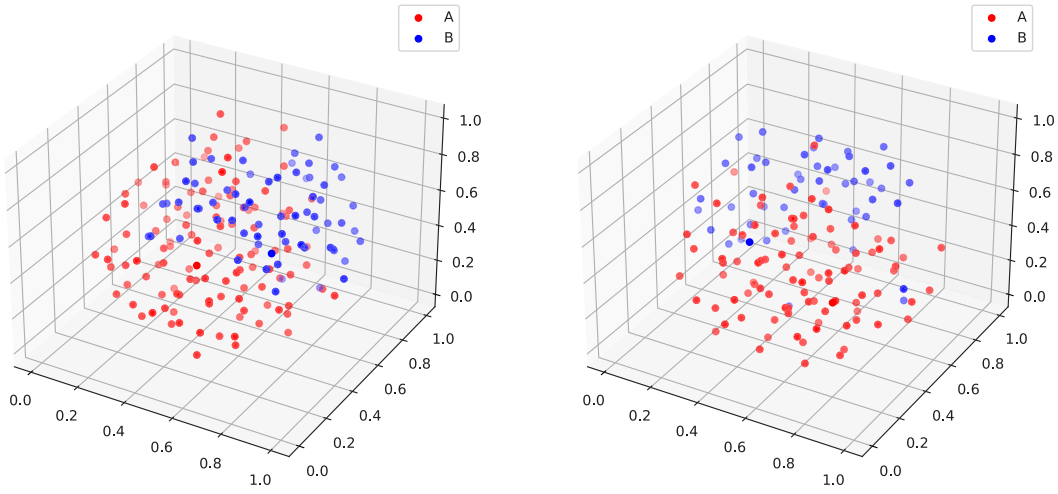


Fig. 13. The model parameters (Param), floating point operations (FLOPs) and training time of our BrainTGL model, vanilla GCN, ASDDiagnet, ST-GCN and GroupINN.



(a) 130 MDD-A and 51 MDD-B (b) 87 BD-A and 59 BD-B

Fig. 14. The clustering result visualization of NMU-MDD dataset and NMU-BD dataset.

has 0.09M more parameters than GCN, but provides more stronger performance. Compared with ASDDiagnet, BrainTGL requires less than 3% parameters of ASDDiagnet and reduced training time. Moreover, BrainTGL also needs only 15% parameters of GroupINN and almost the same training time. The comparison demonstrates that our model achieves higher data utility with less parameters and training time, hence it is more suitable as a spatio-temporal learning model for brain disease analysis, especially when the sample size is limited. Moreover, the FLOPs of ST-GCN is 6850M, showing a high computational complexity of spatio-temporal model in brain analysis of 200 brain regions. Notably, the FLOPs of our model is much lower (only 9.62M) compared to ST-GCN, implying that BrainTGL is a light-weight (i.e., less parameters to be learned) model that is tailored for spatio-temporal brain network with complex structure.

4.5. Disease subtype clustering analysis

In this work, we evaluated our BrainTGL-C on the NMU-MDD dataset and the NMU-BD dataset, respectively. According to the previous study which concludes that there exists two possible subtypes of MDD and BD [31], the cluster number K is set to 2. The cluster results are shown in Fig. 14. We obtained 130 MDD-A, 51 MDD-B for 181 MDD subjects, and 87 BD-A, 59 BD-B for 146 BD subjects through the proposed clustering procedure.

To verify the validity of the clustering results, we conduct two classification tasks of each subtype (subclass) vs. normal controls. The results of two classification tasks are shown in Tables 7 and 8, respectively. In addition, to further evaluate the validity of the subtype identification results, we compute the TN, FN, FP and TP of subtype-A and subtype-B, to obtain the ACC, SEN and SPE of the mixed class MDD(A+B) or BD(A+B) vs. normal controls. The results of MDD(A+B) and BD(A+B) are shown in Tables 7 and 8. It shows that:

(1) It can be seen in Fig. 14 that our BrainTGL-C identifies the inherent subtypes of NMU-MDD and NMU-BD, respectively. The result indicates the effectiveness of our BrainTGL-C.

(2) From the classification result of MDD(A+B) or BD(A+B) vs. normal controls, the deep learning methods GroupINN/BrainnetCNN/ASDDiagnet achieve improvement of 7.8%/3.0%/4.3% and 8.2%/4.5%/3.2% compared with the result of MDD or BD vs. normal controls, respectively. It indicates that the clustered subtypes identified by our BrainTGL-C are helpful for multiple classification models.

5. Limitations and future works

Although compelling, our model still has limitations: (1) we applied the attention based graph pooling to obtain clean and sparse supergraphs containing important connections. However, the pooling operation directly transforms the original graph into a coarsened graph, leading to the loss of the structure information. How to learn a more effective graph structure with maximally preserving the structure information will be the focus of future research. (2) Our model relies on LSTM to capture sufficient temporal features. However, LSTM deals

Table 7
Performance comparison of the various methods on the multiple binary classification tasks of MDD diagnosis on the NMU-MDD dataset. The best value is bolded.

Method	Task	AUC (%)	ACC (%)	SEN (%)	SPE (%)
GroupINN [2]	MDD-A vs. HC	65.3 ± 1.67	69.9 ± 2.69	64.4 ± 2.36	71.3 ± 2.34
	MDD-B vs. HC	**67.1 ± 2.90**	**75.6 ± 1.38**	68.4 ± 0.83	**82.2 ± 2.17**
	MDD(A+B) vs. HC	–	72.3 ± 1.58	66.1 ± 1.69	78.7 ± 1.91
	MDD vs. HC	63.6 ± 1.61	64.5 ± 2.90	**69.4 ± 3.22**	64.8 ± 6.25
BrainnetCNN [27]	MDD-A vs. HC	66.7 ± 2.12	68.9 ± 1.91	67.0 ± 3.34	65.5 ± 2.11
	MDD-B vs. HC	**69.8 ± 2.51**	**72.2 ± 2.14**	**69.7 ± 2.52**	**67.8 ± 2.35**
	MDD(A+B) vs. HC	–	70.9 ± 1.89	69.0 ± 2.95	66.8 ± 2.31
	MDD vs. HC	65.8 ± 1.93	67.9 ± 1.75	67.5 ± 1.63	50.6 ± 2.50
ASDDiagnet [7]	MDD-A vs. HC	66.4 ± 2.05	69.6 ± 1.36	66.0 ± 1.22	75.0 ± 1.87
	MDD-B vs. HC	**68.5 ± 1.94**	**72.8 ± 2.23**	**68.4 ± 2.32**	**75.4 ± 2.18**
	MDD(A+B) vs. HC	–	71.1 ± 1.38	67.9 ± 1.81	75.1 ± 1.67
	MDD vs. HC	62.4 ± 1.74	66.8 ± 1.43	65.9 ± 1.60	69.3 ± 1.41

Table 8
Performance comparison of the various methods on the multiple binary classification tasks of BD diagnosis on the NMU-BD dataset. The best value is bolded.

Method	Task	AUC (%)	ACC (%)	SEN (%)	SPE (%)
GroupINN [2]	BD-A vs. HC	61.5 ± 2.42	73.9 ± 2.59	61.8 ± 2.18	69.6 ± 2.94
	BD-B vs. HC	**63.1 ± 2.73**	**74.4 ± 2.33**	60.4 ± 3.31	**73.2 ± 2.15**
	BD(A+B) vs. HC	–	74.1 ± 2.25	60.9 ± 2.09	71.4 ± 2.21
	BD vs. HC	61.2 ± 2.12	65.9 ± 2.01	**66.1 ± 1.94**	67.1 ± 2.25
BrainnetCNN [27]	BD-A vs. HC	65.6 ± 2.13	72.7 ± 2.92	63.2 ± 2.36	65.4 ± 1.57
	BD-B vs. HC	**66.3 ± 3.24**	**74.0 ± 1.60**	**67.7 ± 2.31**	**68.6 ± 1.63**
	BD(A+B) vs. HC	–	73.3 ± 1.53	66.0 ± 2.17	66.9 ± 1.45
	BD vs. HC	64.5 ± 2.75	68.8 ± 2.57	64.4 ± 1.57	51.4 ± 1.75
ASDDiagnet [7]	BD-A vs. HC	69.6 ± 1.44	73.4 ± 2.56	67.8 ± 3.21	**75.2 ± 3.01**
	BD-B vs. HC	**71.3 ± 2.73**	**74.9 ± 2.61**	**68.6 ± 3.17**	71.4 ± 2.62
	BD(A+B) vs. HC	–	74.4 ± 2.13	68.1 ± 2.56	73.8 ± 2.35
	BD vs. HC	70.1 ± 3.15	71.2 ± 2.36	67.5 ± 2.60	68.3 ± 2.42

with the temporal information in a single direction, which limits our model to sufficiently capture the bidirectional temporal associations in the dynamic brain network series. Therefore, how to capture the long-range dependencies within the temporal graphs is also a future research direction. (3) The proposed BrainTGL have a number of hyperparameters to be empirically chosen along with their architecture. It can be a heavy burden for choosing the architecture and the hyperparameter values for achieving an optimal performance. In most cases, the default hyperparameters and architectures are used. To solve the limitation of the manual hyperparameters and architectures, how to automatically search the optimal hyperparameters and model architectures based on the evolutionary neural architecture search (NAS) will be the focus of future research. Leveraging the neural architecture search methods can exploit the suitable architecture and hyperparameters and further promote the model performance. Therefore, we will conduct more detailed researches on how to improve the spatio-temporal model architecture via NAS methods, including the commonly used computational intelligence algorithms [32–34] such as Monarch Butterfly Optimization (MBO) [35], Earthworm Optimization Algorithm(EWA) [36], Moth Search (MS) [37] algorithm, Slime mould algorithm (SMA) [38], hunger games search (HGS) [39] and so on.

6. Conclusion

In order to achieve a better dynamic graph embedding from brain networks, we develop a temporal graph representation learning model BrainTGL for dynamic brain networks. The proposed BrainTGL model exploits the temporal characteristics in rs-fMRI data through the proposed attention based graph pooling for removing noisy edges and dual temporal graph learning LSTM for learning temporal characteristics in fMRI data from two aspects. Furthermore, we propose an unsupervised spatial–temporal graph clustering framework BrainTGL-C based on our model to analyze the subtypes of brain disease. Extensive experiments were conducted to evaluate the effectiveness of our models on both graph classification and graph clustering tasks, which demonstrates the advantage of our model over the state-of-the-art approaches.

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.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (No. 62076059) and the Science Project of Liaoning province, China (2021-MS-105)

References

[1] D. Yao, M. Liu, M. Wang, C. Lian, J. Wei, L. Sun, J. Sui, D. Shen, Triplet graph convolutional network for multi-scale analysis of functional connectivity using functional MRI, in: International Workshop on Graph Learning in Medical Imaging, Springer, 2019, pp. 70–78.

[2] Y. Yan, J. Zhu, M. Duda, E. Solarz, C. Sripada, D. Koutra, Groupinn: Grouping-based interpretable neural network for classification of limited, noisy brain data, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 772–782.

[3] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: International Conference on Learning Representations (ICLR), 2016, pp. 1–11.

[4] S. Parisot, S.I. Ktena, E. Ferrante, M. Lee, R.G. Moreno, B. Glocker, D. Rueckert, Spectral graph convolutions for population-based disease prediction, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2017, pp. 177–185.

[5] S.I. Ktena, S. Parisot, E. Ferrante, M. Rajchl, M. Lee, B. Glocker, D. Rueckert, Metric learning with spectral graph convolutions on brain connectivity networks, NeuroImage 169 (2018) 431–442.

[6] B. Jie, D. Zhang, C.-Y. Wee, D. Shen, Topological graph kernel on multiple thresholded functional connectivity networks for mild cognitive impairment classification, Hum. Brain Mapp. 35 (7) (2014) 2876–2897.

[7] T. Eslami, V. Mirjalili, A. Fong, A.R. Laird, F. Saeed, ASD-DiagNet: a hybrid learning approach for detection of autism spectrum disorder using fMRI data, Front. Neuroinform. 13 (2019) 70.

[8] R.M. Hutchison, T. Womelsdorf, E.A. Allen, P.A. Bandettini, V.D. Calhoun, M. Corbetta, S. Della Penna, J.H. Duyn, G.H. Glover, J. Gonzalez-Castillo, D. Handwerker, S. Keilholz, V. Kiviniemi, D. Leopold, F. Pasquale, O. Sporns, M. Walter, C. Chang, Dynamic functional connectivity: promise, issues, and interpretations, Neuroimage 80 (2013) 360–378.

[9] M. Wang, C. Lian, D. Yao, D. Zhang, M. Liu, D. Shen, Spatial-temporal dependency modeling and network hub detection for functional MRI analysis via convolutional-recurrent network, IEEE Trans. Biomed. Eng. 67 (8) (2019) 2241–2252.

[10] S. Gadgil, Q. Zhao, A. Pfefferbaum, E.V. Sullivan, E. Adeli, K.M. Pohl, Spatio-temporal graph convolution for resting-state fmri analysis, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2020, pp. 528–538.

[11] T. Azevedo, A. Campbell, R. Romero-Garcia, L. Passamonti, R.A. Bethlehem, P. Liò, N. Toschi, A deep graph neural network architecture for modelling spatio-temporal dynamics in resting-state functional MRI data, Med. Image Anal. 79 (2022) 102471.

[12] K. Ohi, M. Ishibashi, K. Torii, M. Hashimoto, Y. Yano, T. Shioiri, Differences in subcortical brain volumes among patients with schizophrenia and bipolar disorder and healthy controls, J. Psychiatry Neurosci. 47 (2022) 77–85.

[13] M.A. Harris, X. Shen, S.R. Cox, J. Gibson, M.J. Adams, T.-K. Clarke, I.J. Deary, S.M. Lawrie, A.M. McIntosh, H.C. Whalley, et al., Stratifying major depressive disorder by polygenic risk for schizophrenia in relation to structural brain measures, Psychol. Med. 50 (2020) 1653–1662.

[14] X. Liu, L. Li, M. Li, Z. Ren, P. Ma, Characterizing the subtype of anhedonia in major depressive disorder: A symptom-specific multimodal MRI study, Psychiatry Res.: Neuroimaging 308 (2) (2020) 111239.

[15] D.C. Van Essen, S.M. Smith, D.M. Barch, T.E. Behrens, E. Yacoub, K. Ugurbil, W.-M.H. Consortium, The WU-Minn human connectome project: an overview, Neuroimage 80 (2013) 62–79.

[16] A. Manoliu, C. Meng, F. Brandl, A. Doll, M. Tahmasian, M. Scherr, D. Schwerthöffer, C. Zimmer, H. Förstl, J.G. Bäuml, V. Riedl, A.M. Wohlschläger, C. Sorg, The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism, Mol. Psychiatry 19 (6) (2014) 659–667.

[17] J. Ji, X. Xing, Y. Yao, J. Li, X. Zhang, Convolutional kernels with an element-wise weighting mechanism for identifying abnormal brain connectivity patterns, Pattern Recognit. 109 (2021) 107570.

[18] J.F.A. Ronicko, J. Thomas, P. Thangavel, V. Koneru, G. Langs, J. Dauwels, Diagnostic classification of autism using resting-state fMRI data improves with full correlation functional brain connectivity compared to partial correlation, J. Neurosci. Methods 345 (2020) 108884.

[19] R.M. Thomas, S. Gallo, L. Cerliani, P. Zhutovsky, A. El-Gazzar, G. Van Wingen, Classifying autism spectrum disorder using the temporal statistics of resting-state functional MRI data with 3D convolutional neural networks, Front. Psychiatry 11 (2020) 440.

[20] X. Li, Y. Zhou, N. Dvornek, M. Zhang, S. Gao, J. Zhuang, D. Scheinost, L.H. Staib, P. Ventola, J.S. Duncan, Braingnn: Interpretable brain graph neural network for fmri analysis, Med. Image Anal. 74 (2021) 102233.

[21] L. Li, H. Jiang, G. Wen, P. Cao, M. Xu, X. Liu, J. Yang, O. Zaiane, TE-HI-GCN: An ensemble of transfer hierarchical graph convolutional networks for disorder diagnosis, Neuroinformatics (2021) 1–23.

[22] R.P. Monti, P. Hellyer, D. Sharp, R. Leech, C. Anagnostopoulos, G. Montana, Estimating time-varying brain connectivity networks from functional MRI time series, NeuroImage 103 (2014) 427–443.

[23] B. Jie, M. Liu, D. Shen, Integration of temporal and spatial properties of dynamic connectivity networks for automatic diagnosis of brain disease, Med. Image Anal. 47 (2018) 81–94.

[24] M. Wang, J. Huang, M. Liu, D. Zhang, Modeling dynamic characteristics of brain functional connectivity networks using resting-state functional MRI, Med. Image Anal. 71 (2021) 102063.

[25] K. Lin, B. Jie, P. Dong, X. Ding, W. Bian, M. Liu, Convolutional recurrent neural network for dynamic functional MRI analysis and brain disease identification, Front. Neurosci. (2022) 1050.

[26] Z. Yin, K. Yue, Temporal resonant graph network for representation learning on dynamic graphs, Appl. Intell. (2022) 1–18.

[27] J. Kawahara, C. Brown, S. Miller, B. Booth, V. Chau, R. Grunau, J. Zwicker, G. Hamarneh, BrainNetCNN: Convolutional neural networks for brain networks; towards predicting neurodevelopment, NeuroImage 146 (2016).

[28] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, J. Leskovec, Hierarchical graph representation learning with differentiable pooling, Adv. Neural Inf. Process. Syst. 31 (2018).

[29] J. Lee, I. Lee, J. Kang, Self-attention graph pooling, in: International Conference on Machine Learning, PMLR, 2019, pp. 3734–3743.

[30] J. Huang, Z. Li, N. Li, S. Liu, G. Li, AttPool: Towards hierarchical feature representation in graph convolutional networks via attention mechanism, in: 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 6479–6488.

[31] M. Chang, F.Y. Womer, X. Gong, X. Chen, L. Tang, R. Feng, S. Dong, J. Duan, Y. Chen, R. Zhang, et al., Identifying and validating subtypes within major psychiatric disorders based on frontal–posterior functional imbalance via deep learning, Mol. Psychiatry 26 (7) (2021) 2991–3002.

[32] G. Li, G.-G. Wang, J. Dong, W.-C. Yeh, K. Li, DLEA: A dynamic learning evolution algorithm for many-objective optimization, Inform. Sci. 574 (2021) 567–589.

[33] Y. Zhang, G.-G. Wang, K. Li, W.-C. Yeh, M. Jian, J. Dong, Enhancing MOEA/D with information feedback models for large-scale many-objective optimization, Inform. Sci. 522 (2020) 1–16.

[34] Z.-M. Gu, G.-G. Wang, Improving NSGA-III algorithms with information feedback models for large-scale many-objective optimization, Future Gener. Comput. Syst. 107 (2020) 49–69.

[35] G.-G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization, Neural Comput. Appl. 31 (7) (2019) 1995–2014.

[36] G.-G. Wang, S. Deb, L.D.S. Coelho, Earthworm optimisation algorithm: a bio-inspired metaheuristic algorithm for global optimisation problems, Int. J. Bio-Inspired Comput. 12 (1) (2018) 1–22.

[37] G.-G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, Memet. Comput. 10 (2) (2018) 151–164.

[38] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: A new method for stochastic optimization, Future Gener. Comput. Syst. 111 (2020) 300–323.

[39] Y. Yang, H. Chen, A.A. Heidari, A.H. Gandomi, Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts, Expert Syst. Appl. 177 (2021) 114864.