BrainGNN: Interpretable Brain Graph Neural Network for fMRI Analysis*

Xiaoxiao Li[1†], Yuan Zhou[4], Siyuan Gao[1‡], Nicha Dvornek[1,4‡], Muhan Zhang[6‡], Juntang Zhuang[1], Shi Gu[5], Dustin Scheinost[4], Lawrence Staib[1,3,4], Pamela Ventola[2], and James Duncan[1,3,4]

[1] Biomedical Engineering, Yale University, New Haven, CT, 06511, USA
[2] Child Study Center, Yale School of Medicine, New Haven, CT, 06511, USA
[3] Electrical Engineering, Yale University, New Haven, CT, 06511, USA
[4] Radiology & Biomedical Imaging, Yale School of Medicine, New Haven, CT, 06511, USA
[5] Department of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China
[6] Facebook AI

Abstract. Understanding how certain brain regions relate to a specific neurological disorder or cognitive stimuli has been an important area of neuroimaging research. We propose BrainGNN, a graph neural network (GNN) framework to analyze functional magnetic resonance images (fMRI) and discover neurological biomarkers. In contrast to feedforward neural networks (FNN) and convolutional neural networks (CNN) in traditional functional connectivity-based fMRI analysis methods, we construct weighted graphs from fMRI and apply a GNN to fMRI brain graphs. Considering the special property of brain graphs, we design novel brain ROI-aware graph convolutional layers (Ra-GNN) that leverages the topological and functional information of fMRI. Motivated by the need for transparency in medical image analysis, our BrainGNN contains ROI-selection pooling layers (R-pool) that highlight salient ROIs (nodes in the graph), so that we can infer which ROIs are important for prediction. Furthermore, we propose regularization terms - unit loss, topK pooling (TPK) loss and group-level consistency (GLC) loss - on pooling results to encourage reasonable ROI-selection and provide flexibility to preserve either individual- or group-level patterns. We apply the BrainGNN framework on two independent fMRI datasets: Autism Spectral Disorder (ASD) fMRI dataset and Human Connectome Project (HCP) 900 Subject Release. We investigate different choices of the hyperparameters and show that BrainGNN outperforms the alternative FNN, CNN and GNN-based fMRI image analysis methods in terms of classification accuracy. The obtained community clustering and salient ROI detection results show high correspondence with the previous neuroimaging-derived evidence of biomarkers for ASD and specific task states decoded in task-fMRI.

*In submission

†Corresponding Author: Xiaoxiao Li, xiaoxiao.li@aya.yale.edu

‡Equal contribution

Keywords: Graph Neural Network, Neuroimaging, Interpretability

1 Introduction

The brain is an exceptionally complex system and understanding it's functional organization is the goal of modern neuroscience. Using fMRI, large strides in understanding this organization have been made by modeling the brain as a graph—a mathematical construct describing the connections or interactions (i.e. edges) between different discrete objects (i.e. nodes). To create these graphs, nodes are defined as brain regions of interest (ROIs) and edges are defined as the functional connectivity between those ROIs, computed as the pairwise correlations of functional magnetic resonance imaging (fMRI) time series, as illustrated in Fig. 1. Traditional graph-based analyses for fMRI have focused on using graph theoretical metrics to summarize the functional connectivity for each node into a single number [46,23]. However, these methods do not consider higher-order interactions between ROIs, as these interactions cannot be preserved in a single number. Additionally, due to the high dimensionality of fMRI data, usually ROIs are clustered into highly connected communities to reduce dimensionality. Then, features are extracted from these smaller communities for further analysis [32,12]. For these two-stage methods, if the results from the first stage are not reliable, significant errors can be induced in the second stage.

The past few years have seen the growing prevalence of the use of graph neural networks (GNN) for end-to-end graph learning applications. GNNs are the state-of-the-art deep learning methods for most graph-structured data analysis problems. They combine node features, edge features, and graph structure by using a neural network to embed node information and pass information through edges in the graph. As such, they can be viewed as a generalization of the traditional convolutional neural networks (CNN) for images. Due to their high performance and interpretability, GNNs have been a widely applied graph analysis method. [26,25,50,28,51]. Most existing GNNs are built on graphs that do not have correspondence between the nodes of different instances, such as social networks and protein networks, limiting interpretability. These methods – including the current GNN methods for fMRI analysis – use the same kernel over different nodes, which implicitly assumes brain graphs are translation invariant. However, nodes in the same brain graph have distinct locations and unique identities. Thus, applying the same kernel over all nodes is problematic. In addition, few GNN studies have explored both individual-level and group-level explanations, which are critical in neuroimaging research.

In this work, we propose a graph neural network-based framework for mapping regional and cross-regional functional activation patterns for classification tasks, such as classifying neurodisorder patients versus healthy control subjects and performing cognitive task decoding. Our framework jointly learns ROI clustering and the downstream whole-brain fMRI analysis. This not only reduces preconceived errors, but also learns particular clustering patterns associated with the downstream tasks. Specifically, from estimated model parameters, we can

retrieve ROI clustering patterns. Also, our GNN design facilitates model interpretability by regulating intermediate outputs with a novel loss term, which provides the flexibility to choose between individual-level and group-level explanations.

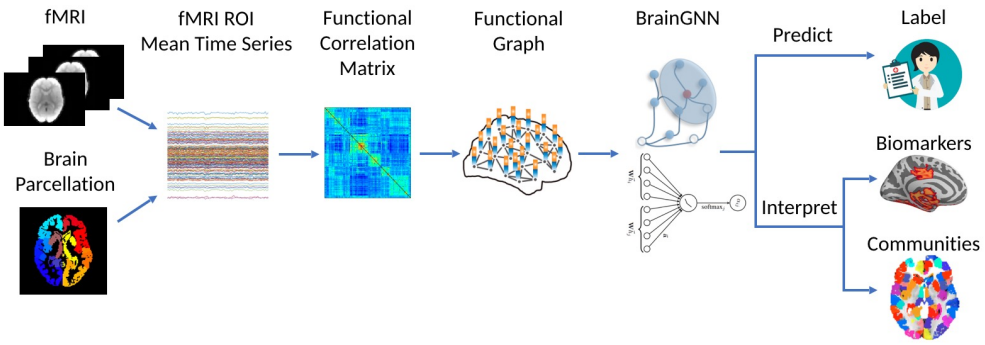


Fig. 1: The overview of the pipeline. fMRI images are parcellated by an atlas and transferred to graphs. Then, the graphs are sent to our proposed BrainGNN, which gives the prediction of specific tasks. Jointly, BrainGNN selects salient brain regions that are informative to the prediction task and clusters brain regions into prediction-related communities.

2 Methods and Materials

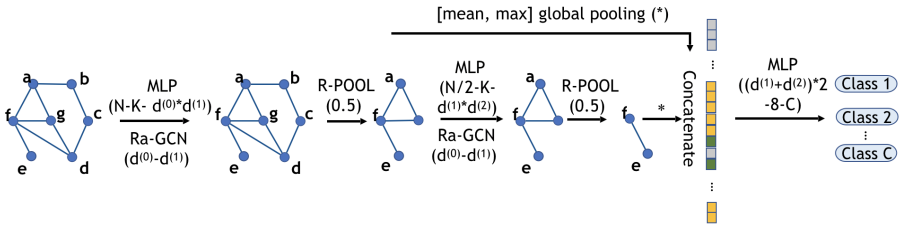
2.1 Preliminaries

Notation and Problem Definition

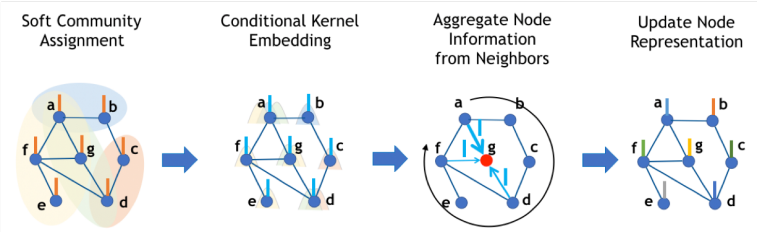
First we parcellate the brain into N regions of interest (ROIs) based on its T1 structural MRI. We define ROIs as graph nodes $\mathcal{V} = \{v_1, \ldots, v_N\}$ and the nodes are preordered. As brain ROIs can be aligned by brain parcellation atlases based on their location in the structure space, we define the brain graphs as ordered aligned graphs. We define an undirected weighted graph as $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{E}$ is the edge set, i.e., a collection of (v_i, v_j) linking vertices from v_i to v_j. In our setting, G has an associated node feature set $\mathcal{H} = \{\mathbf{h}_1, \ldots, \mathbf{h}_N\}$, where $\mathbf{h}_i$ is the feature vector associated with node v_i. For every edge connecting two nodes, $(v_i, v_j) \in \mathcal{E}$, we have its strength $e_{ij} \in \mathbb{R}$ and $e_{ij} > 0$. We also define $e_{ij} = 0$ for $(v_i, v_j) \notin \mathcal{E}$ and therefore the adjacency matrix $E = [e_{ij}] \in \mathbb{R}^{N \times N}$ is well defined.

Architecture Overview

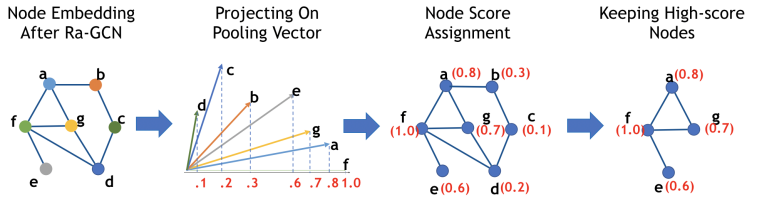
Classification on graphs is achieved by first embedding node features into a low-dimensional space, then grouping nodes and summarizing them. The summarized



(a) BrainGNN Architecture



(b) Operations in Ra-GNN layer



(c) Operations in R-pool Layer

Fig. 2: (a) introduces the BrainGNN architecture that we propose in this work. BrainGNN is composed of Ra-GNN and R-pool blocks. It takes graphs as inputs and outputs graph-level predictions. (b) shows how the Ra-GNN layer embeds node features. First, nodes are softly assigned to communities based on their membership scores to the communities. Each community is associated with a different basis vector. Each node is embedded by the particular basis vectors based on the communities that it belongs to. Then, by aggregating a node's own embedding and its neighbors' embedding, the updated representation is assigned to each node on the graph. (c) shows how R-pool selects nodes to keep. First, all the nodes' representations are projected to a learnable vector. The nodes with large projected values are retained with their corresponding connections.

vector is then fed into a classifier, such as a multilayer perceptron (MLP), potentially in an end-to-end fashion. Our proposed network architecture is illustrated in Figure 2a. It is formed by three different types of layers: graph convolutional layers, node pooling layers and a readout layer. Generally speaking, GNNs inductively learn a node representation by recursively transforming and aggregating the feature vectors of its neighboring nodes.

A **graph convolutional layer** is used to probe the graph structure by using edge features, which contain important information about graphs. For example, the weights of the edges in brain fMRI graphs can represent the relationship between different ROIs.

Following [39], we define $\mathbf{h}_i^{(l)} \in \mathbb{R}^{d^{(l)}}$ as the features for the i^{th} node in the l^{th} layer, where $d^{(l)}$ is the dimension of the l^{th} layer features. The propagation model for the forward-pass update of node representation is calculated as:

$$\mathbf{h}_i^{(l)} = \sigma\left(W_0^{(l-1)}\mathbf{h}_i^{(l-1)} + \sum_{j\in\mathcal{N}(i)} \phi\left(W_1^{(l-1)}\mathbf{h}_j^{(l-1)}, e_{ij}\right)\right), \tag{1}$$

where $\mathcal{N}(i)$ denotes the set of indices of neighboring nodes of node v_i and e_{ij} denotes the features associated with the edge from v_i to v_j, W_0, W_1 denote the model's parameters to be learned, and ϕ is any linear/nonlinear function that can be applied on the neighboring nodes' feature embedding. σ is the activation function.

A **node pooling** layer is used to reduce the size of the graph, either by grouping the nodes together or pruning the original graph G to a subgraph G_s by keeping some important nodes only. We will focus on the pruning method, as it is more interpretable and can help detect biomarkers.

A **readout** layer is used to summarize the node feature vectors $\{\mathbf{h}_i^{(l)}\}$ into a single vector $\mathbf{z}$ which is finally fed into a classifier for graph classification.

2.2 Proposed Approach

In this section, we provide insights and highlight the innovative design aspects of our proposed BrainGNN architecture.

ROI-aware Graph Convolutional Layer

Overview We propose an ROI-aware graph convolutional neural network (Ra-GNN) with two insights. First, when computing the node embedding, we allow Ra-GNN to learn different convolutional kernels conditioned on the ROI (geometric information of the brain), instead of using the same kernel W on all the nodes as it is shown in Eq. (1). Second, we include edge weights for message filtering, as the magnitude of edge weights presents the connection strength between two ROIs. We assume more closely connected ROIs have higher impact.

Design We begin by assuming the graphs have additional regional information and the nodes of the same region from different graphs have similar properties. We propose to encode the regional information to the embedding kernel function for the nodes. Given node i's regional information $\mathbf{r}_i$, such as the node's coordinates in a mesh graph, we propose to learn the vectorized embedding kernel $\text{vec}(W_i)$ based on $\mathbf{r}_i$ on the l^{th} Ra-GNN:

$$\text{vec}(W_i^{(l)}) = f_{MLP^{(l)}}(\mathbf{r}_i) = \Theta_2^{(l)}\text{relu}(\Theta_1^{(l)}\mathbf{r}_i) + \mathbf{b}^{(l)}, \tag{2}$$

where the MLP network with parameters $\{\Theta_1^{(l)}, \Theta_2^{(l)}\}$ maps $\mathbf{r}_i$ to a $d^{(l)} \cdot d^{(l-1)}$ dimensional vector then reshapes the output to a $d^{(l)} \times d^{(l-1)}$ matrix $W_i^{(l)}$.

Given a brain parcellated into N ROIs, we order the ROIs in the same manner for all the brain graphs. Therefore, the nodes in the graphs of different subjects are aligned. However, the convolutional embedding should be independent of the ordering methods. Given an ROI ordering for all the graphs, we use one-hot encoding to represent the ROI's location information, instead of using coordinates, because the nodes in the brain are aligned well. Specifically, for node v_i, its ROI representation $\mathbf{r}_i$ is a $N-$dimensional vector with 1 in the i^{th} entry and 0 for the other entries. Assume that $\Theta_1^{(l)} = [\boldsymbol{\alpha}_1^{(l)}, \ldots, \boldsymbol{\alpha}_{N^{(l)}}^{(l)}]$, where $N^{(l)}$ is the number of ROIs left on the l^{th} layer, $\boldsymbol{\alpha}_i^{(l)} = [\alpha_{i1}^{(l)}, \ldots, \alpha_{iK^{(l)}}^{(l)}]^T \in \mathbb{R}^{K^{(l)}}$, $\forall i \in \{1, \ldots, N^{(l)}\}$, where $K^{(l)}$ can be seen as the number of clustered communities for the $N^{(l)}$ ROIs. Assume $\Theta_2^{(l)} = [\boldsymbol{\beta}_1^{(l)}, \ldots, \boldsymbol{\beta}_{K^{(l)}}^{(l)}]$ with $\boldsymbol{\beta}_j^{(l)} \in \mathbb{R}^{d^{(l)} \cdot d^{(l-1)}}$, $\forall j \in \{1, \ldots, K^{(l)}\}$. Then Eq. (2) can be rewritten as

$$\mathrm{vec}(W_i^{(l)}) = \sum_{j=1}^{K^{(l)}} (\alpha_{ij}^{(l)})^+ \boldsymbol{\beta}_j^{(l)} + \boldsymbol{b}^{(l)}. \tag{3}$$

We can view $\{\boldsymbol{\beta}_j^{(l)} : j = 1, \ldots, K^{(l)}\}$ as a basis and $(\alpha_{ij}^{(l)})^+$ as the coordinates. From another perspective, $(\alpha_{ij}^{(l)})^+$ can been seen as the non-negative assignment score of ROI i to community j. If we train different embedding kernels for different ROIs on l^{th} Ra-GNN, the total parameters to be learned will be $N^{(l)} d^{(l)} d^{(l-1)}$. Usually we have $K^{(l)} \ll N^{(l)}$. By Eq. (3), we can reduce the number of learnable parameters to $K^{(l)} d^{(l)} d^{(l-1)} + N^{(l)} K^{(l)}$ parameters, while still assigning a separate embedding kernel for each ROI. The ROIs in the same community will be embedded by the similar kernel so that nodes in different communities are embedded in different ways.

As the graph convolution operations in [17], the node features will be multiplied by the edge weights, so that neighbors connected with stronger edges have a larger influence. The GNN layer using ROI-aware kernels and edge weights for filtering can be written as:

$$\mathbf{h}_i^{(l)} = W_i^{(l-1)} \mathbf{h}_i^{(l-1)} + \sum_{j \in \mathcal{N}(i)} \tilde{e}_{ij} W_j^{(l-1)} \mathbf{h}_j^{(l-1)}, \tag{4}$$

To avoid increasing the scale of output features, the edge features need to be normalized, as in GAT [43] and GNN [27]. Due to the aggregation mechanism, we normalize the weights by $\tilde{e}_{ij} = e_{ij} / \sum_{j \in \mathcal{N}(i)} e_{ij}$.

ROI-topK Pooling Layer

Overview To perform graph-level classification, a layer for dimensionality reduction is needed since the number of nodes and the feature dimension per node are both large. Recent findings have shown that some ROIs are more indicative of predicting neurological disorders than the others [22,2], suggesting that they

should be kept in the dimensionality reduction step. Therefore the node (ROI) pooling layer (R-pool) is designed to keep the most indicative ROIs, thereby reducing dimensionality and removing *noisy* nodes.

Design To make sure that down-sampling layers behave idiomatically with respect to different graph sizes and structures, we adopt the approach in [6,15] for reducing graph nodes. The choice of which nodes to drop is determined based on projecting the node attributes onto a learnable vector $\mathbf{w}^{(l-1)} \in \mathbb{R}^{d^{(l-1)}}$. The nodes receiving lower scores will experience less feature retention. We denote $H^{(l-1)} = [\mathbf{h}_1^{(l-1)}, \ldots, \mathbf{h}_{N^{(l-1)}}^{(l-1)}]^T$, where $N^{(l-1)}$ is the number of nodes at the $(l-1)^{th}$ layer. Fully written out, the operation of this pooling layer (computing a pooled graph, $(\mathcal{V}^{(l)}, \mathcal{E}^{(l)})$, from an input graph, $(\mathcal{V}^{(l-1)}, \mathcal{E}^{(l-1)})$), is expressed as follows:

$$\begin{aligned} \mathbf{s}^{(l-1)} &= H^{(l-1)}\mathbf{w}^{(l-1)}/\|\mathbf{w}^{(l-1)}\| \\ \tilde{\mathbf{s}}^{(l-1)} &= (\mathbf{s}^{(l-1)} - \mu(\mathbf{s}^{(l-1)}))/\sigma(\mathbf{s}^{(l-1)}) \\ \mathbf{i} &= \text{top}k(\tilde{\mathbf{s}}^{(l-1)}, k) \\ H^{(l)} &= (H^{(l-1)} \odot \text{sigmoid}(\tilde{\mathbf{s}}^{(l-1)}))_{\mathbf{i},:} \\ E^{(l)} &= E^{(l-1)}_{\mathbf{i},\mathbf{i}}. \end{aligned} \tag{5}$$

Here $\|\cdot\|$ is the L_2 norm, μ and σ take the input vector and output the mean and standard deviation of its elements. The notation top$k$ finds the indices corresponding to the largest k elements in score vector $\mathbf{s}$, $\odot$ is (broadcasted) element-wise multiplication, and $(\cdot)_{\mathbf{i},\mathbf{j}}$ is an indexing operation which takes elements at row indices specified by $\mathbf{i}$ and column indices specified by $\mathbf{j}$ (colon denotes all indices). The pooling operation retains sparsity by requiring only a projection, a point-wise multiplication and a slicing into the original features and adjacency matrix. Different from [6], we induce the constraint $\|\mathbf{w}^{(l)}\|_2 = 1$ implemented by adding an additional regularization loss $\sum_{l=1}^{L}(\|\mathbf{w}^{(l)}\|_2 - 1)^2$ to avoid identifiability issues. In addition, we added element-wise score normalization $\tilde{\mathbf{s}}^{(l)} = (\mathbf{s}^{(l)} - \mu(\mathbf{s}^{(l)}))/\sigma(\mathbf{s}^{(l)})$, which is important for calculating the GLC loss and TPK loss (introduced in Section 2.3).

Readout Layer

Lastly, we seek a "flattening" operation to preserve information about the input graph in a fixed-size representation. Concretely, to summarize the output graph of the lth conv-pool block, $(\mathcal{V}^{(l)}, \mathcal{E}^{(l)})$, we use

$$\mathbf{z}^{(l)} = \text{mean}\, \mathcal{H}^{(l)} \parallel \max \mathcal{H}^{(l)}, \tag{6}$$

where $\mathcal{H}^{(l)} = \{\mathbf{h}_i^{(l)} : i = 1, ..., N^{(l)}\}$, mean and max operate elementwisely, and $\parallel$ denotes concatenation. To retain information of a graph in a vector, we concatenate both mean and max summarization for a more informative graph-level representation. The final summary vector is obtained as the concatenation of all those summaries (i.e. $\mathbf{z} = \mathbf{z}^{(1)} \parallel \mathbf{z}^{(2)} \parallel \cdots \parallel \mathbf{z}^{(L)}$) and it is submitted to a MLP for obtaining final predictions.

2.3 Putting Layers Together and Loss Functions

All in all, the architecture (as shown in Fig. 2a) consists of two kinds of layers. The input is the weighted graph with its node attributes constructed from fMRI. We form a two-layer GNN block starting with ROI-aware node embedding by the proposed Ra-GNN layer in Section 2.2, followed by the proposed R-pool layer in Section 2.2. The whole network sequentially concatenates these GNN blocks, and readout layers are added after each GNN block. The final summary vector concatenates all those summaries, and an MLP is applied after that to give final predictions. Now we describe the loss function for the neural network.

The classification loss is the cross entropy loss:

$$L_{ce} = -\frac{1}{M}\sum_{m=1}^{M}\sum_{c=1}^{C} y_{m,c}\log(\hat{y}_{m,c}), \tag{7}$$

where M is the number of instances, C is the number of classes, y_{mc} is the ground truth label and $\hat{y}_{mc}$ is the model output.

We add several loss terms to regulate the learning process and control the interpretability. First, as we mentioned in Section 2.2, to avoid the problem of identifiability, we propose unit loss:

$$L_{unit}^{(l)} = (\|\mathbf{w}^{(l)}\|_2 - 1)^2. \tag{8}$$

Note that $\tilde{\mathbf{s}}^{(l)}$ in Eq. (5) is computed from the input $H^{(l)}$. Therefore, for different inputs $H^{(l)}$, the selected entries of $\tilde{\mathbf{s}}^{(l)}$ can be very different. For our application, we want to find the common patterns/biomarkers for a certain neuro-prediction task. Thus, we add regularization to force the $\tilde{\mathbf{s}}^{(l)}$ vectors to be similar for different input instances after the first pooling layer and call it group-level consistency (GLC). We do not constrain GLC for the second pooling layer because the nodes after the first pooling layer in different graphs might be different.

In each training batch, suppose there are M instances, which can be partitioned into C subsets based on the class labels, $\mathcal{I}_c = \{m : m = 1, \ldots, M, y_{m,c} = 1\}$, for $c = 1, \ldots, C$. And $y_{m,c} = 1$ indicates the m^{th} instance belonging to class c. We form the scoring matrix for the instances belonging to class c as $S_c^{(1)} = [\tilde{\mathbf{s}}_i^{(1)} : i \in \mathcal{I}_c]^T \in \mathbb{R}^{M_c \times N}$, where $M_c = |\mathcal{I}_c|$. The GLC loss can be expressed as:

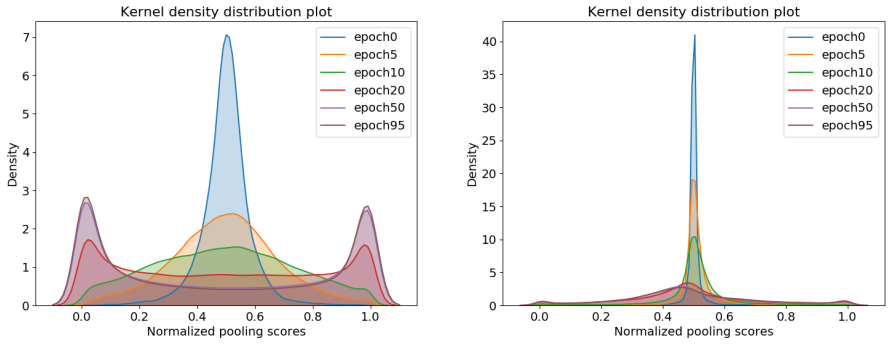
$$\begin{aligned} L_{GLC} &= \sum_{c=1}^{C}\sum_{i,j\in\mathcal{I}_c} \| \tilde{\mathbf{s}}_i^{(1)} - \tilde{\mathbf{s}}_j^{(1)} \|^2 \\ &= \sum_{c=1}^{C} \left\{ 2\mathrm{Tr}((S_c^{(1)})^T D_c S_c^{(1)}) - 2\mathrm{Tr}((S_c^{(1)})^T W_c S_c^{(1)}) \right\} \\ &= 2\sum_{c=1}^{C} \mathrm{Tr}((S_c^{(1)})^T L_c S_c^{(1)}), \end{aligned} \tag{9}$$

where W_c is a $M_c \times M_c$ matrix with all 1s, D_c is a $M_c \times M_c$ diagonal matrix with M_c as diagonal elements, and $L_c = D_c - W_c$ is a symmetric positive semidefinite matrix [45].

In addition, we hope the top k selected indicative ROIs should have significantly different scores than those of the unselected nodes. Ideally, the scores for the selected nodes should be close to 1 and the scores for the unselected nodes should be close to 0. To achieve this, we rank sigmoid($\tilde{\mathbf{s}}_m^{(l)}$) for the mth instance in a descending order, denote it as $\hat{\mathbf{s}}_m^{(l)} = [\hat{s}_{m,1}^{(l)}, \ldots, \hat{s}_{m,N^{(l)}}^{(l)}]$, and apply a constraint to all the M training instances to make the values of $\hat{\mathbf{s}}_m^{(l)}$ more dispersed. We define TPK loss using binary cross-entropy as:

$$L_{TPK}^{(l)} = -\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N^{(l)}}\Big(\sum_{i=1}^{k}\log(\hat{s}_{m,i}^{(l)}) + \sum_{i=1}^{N^{(l)}-k}\log(1-\hat{s}_{m,i+k}^{(l)})\Big), \tag{10}$$

We show the kernel density estimate plots of normalized node pooling scores (indication of the importance of the nodes) changing over the training epoch in Fig. 3 when $k = \frac{1}{2}N^{(l)}$. It is clear to see that the pooling scores are more dispersed over time, Hence the top 50% selected nodes have significantly higher importance scores than the unselected ones.



(a) The change of the distribution of node pooling scores $\hat{\mathbf{s}}$ of the 1st R-pool layer over 100 training epochs.

(b) The change of the distribution of node pooling scores $\hat{\mathbf{s}}$ of the 2nd R-pool layer over 100 training epochs.

Fig. 3: Effect of TopK pooling (TPK) loss. With the TPK loss regularization, the node pooling scores of the selected nodes and those of the unselected nodes become significantly separate.

Finally, the final loss function is formed as:

$$L_{total} = L_{ce} + \sum_{l=1}^{L} \lambda_1^{(l)} L_{unit}^{(l)} + \sum_{l=1}^{L} \lambda_2^{(l)} L_{TPK}^{(l)} + \lambda_3 L_{GLC}, \tag{11}$$

where λ's are tunable hyper-parameters, l indicates the l^{th} GNN block and L is the total number of GNN blocks. GLC loss is only calculated based on the pooling scores of the first pooling layer.

2.4 Interpretation from BrainGNN

Community Detection from Convolutional Layers The important contribution of our proposed ROI-aware convolutional layer is the implied community clustering patterns in the graph. Discovering brain community patterns is critical to understanding co-activation and interaction in the brain. Revisiting Eq. (3), α_{ij}^{+} provides the membership of ROI i to community j. The community assignment is soft and overlaid. It is similar to tensor decomposition-based community detection methods, such as PARAFAC [7], that decompose the tensor to discover overlapping functional brain networks. Parameter $\boldsymbol{\alpha}_i^{+}$ can be seen as the loading vector in PARAFAC that presents the membership of each node to a certain community. Hence, we consider region i belongs to community j if $\alpha_{ij} > \mu(\boldsymbol{\alpha}_i^{+}) + \sigma(\boldsymbol{\alpha}_i^{+})$ [29]. This gives us a collection of community indices indicating region membership $\{\boldsymbol{i}_j \subset \{1, ..., N\} : j = 1, ..., K\}$.

Biomarker Detection from Pooling Layers Without the added TPK loss (Eq. (10)), the significance of the nodes left after pooling cannot be guaranteed. With TPK loss, pooling scores are more dispersed over time, hence the selected nodes have significantly higher importance scores than the unselected ones. The strength of the GLC loss controls the tradeoff between individual-level interpretation and group-level interpretation. On the one hand, for precision medicine, individual-level biomarkers are desired for planning targeted treatment. On the other hand, group-level biomarkers are essential for understanding the common characteristic patterns associated with the disease. We can tune the coefficient λ_3 to control different levels of interpretation. Large λ_3 encourages selecting similar nodes, while small λ_3 allows various node selection results for different instances.

2.5 Datasets

Two independent datasets, the Biopoint Autism Study Dataset (Biopoint) [44] and the Human Connectome Project (HCP) 900 Subject Release [42], are used in this work. For the Biopoint dataset, the aim is to classify Autism Spectrum Disorder (ASD) and Healthy Control (HC). For the HCP dataset, the aim is to classify 7 task states - gambling, language, motor, relational, social, working memory (WM), emotion.

Biopoint Dataset The Biopoint Autism Study Dataset [44] contains 72 ASD children and 43 age-matched ($p > 0.124$) and IQ-matched ($p > 0.122$) neurotypical healthy controls (HCs). For the fMRI scans, subjects perform the "biopoint" task, viewing point-light animations of coherent and scrambled biological motion in a block design [22] ($24s$ per block).

The fMRI data are preprocessed using FSL as follows: 1) motion correction using MCFLIRT, 2) interleaved slice timing correction, 3) BET brain extraction, 4) spatial smoothing (FWHM=5mm), and 5) high-pass temporal filtering. The functional and anatomical data are registered to the MNI152 standard brain atlas [44] using FreeSurfer. The first few frames are discarded, resulting in 146 frames for each fMRI sequence.

The Desikan-Killiany [11] atlas is used to parcellate brain images into 84 ROIs. The mean time series for each node is extracted from a random 1/3 of voxels in the ROI (given an atlas) by bootstrapping. We augment the data 30 times, resulting in 2160 ASD graphs and 1290 HC graphs separately. Edges are defined by thresholding (top 10% positive) partial correlations to achieve sparse connections. For node attributes, we concatenate seven handcrafted features: the degree of node, the mean and standard deviation of the task-fMRI time series, General Linear Model (GLM) coefficients, and Pearson correlation coefficient to node $1-84$. Pearson correlation and partial correlation are different measures of fMRI connectivity. We aggregate them by using one to build edge connections and the other to build node features. For the GLM coefficients, they are the coefficients of the biological motion matrix, the coefficient of the scramble motion matrix, and the coefficients of the previous two matrices' derivatives in the "biopoint task". Hence, node feature $\mathbf{h}_i^{(0)} \in \mathbb{R}^{(7+84)}$.

HCP Dataset For this dataset, we restrict our analyses to those individuals who participated in all nine fMRI conditions (seven tasks, two rests) with full length of scan, whose mean frame-to-frame displacement is less than 0.1 mm and whose maximum frame-to-frame displacement is less than 0.15 mm (n=506; 237 males; ages 22–37). This conservative threshold for exclusion due to motion is used to mitigate the substantial effects of motion on functional connectivity; only left-right (LR) phase encoding run data are considered.

fMRI data were processed with standard methods (see [14] for more details) and parcellated into 268 nodes using a whole-brain, functional atlas defined in a separate sample (see [18] for more details). Task functional connectivity was calculated based on the raw task time series: the mean time series of each node pair were used to calculate the Pearson correlation and partial correlation. Matrices were generated for LR phase encoding runs in the HCP data, and these matrices were averaged for each condition, thus generating one 268×268 Pearson correlation connectivity matrix and partial correlation connectivity matrix per individual per task condition. We define a weighted undirected graph with 268 nodes per individual per task condition resulting in 3542 graphs in total. The same graph construction method as for the Biopoint data was used: nodes represent parcellated brain regions, and edges are constructed by thresholding (top 10% positive)

partial correlation. For node attributes, we concatenate three handcrafted features: degree of node, mean and standard deviation of task-fMRI time series, and Pearson correlation coefficient to node $1-268$, as GLM parameters are not useful for task state classification. Hence, node feature $\mathbf{h}_i^{(0)} \in \mathbb{R}^{(3+268)}$.

2.6 Implementation Details

We trained and tested the algorithm on Pytorch in the Python environment using a NVIDIA Geforce GTX 1080Ti with 11GB GPU memory. The model architecture was implemented with 2 conv layers and 2 pooling layers as shown in Fig. 2a, with parameter $N = 84, K^{(1)} = K^{(2)} = 8, d^{(0)} = 91, d^{(1)} = 16, d^{(2)} = 16, C = 2$ for the Biopoint dataset and $N = 268, K^{(1)} = K^{(2)} = 8, d^{(0)} = 271, d^{(1)} = 32, d^{(2)} = 32, C = 7$ for HCP dataset. The pooling ratio is 0.5. $\lambda_1^{(1)}$ and $\lambda_1^{(2)}$ were fixed to 1. The motivation of $K = 8$ comes from the eight functional networks defined by Finn et al. [14].

We will discuss the variation of $\lambda_2^{(1)}, \lambda_2^{(2)}$ and λ_3 in Section 3.1. We randomly split the data into five folds based on subjects, which means that the graphs from a single subject can only appear in either the training or testing dataset. Four folds were used as training data, and the left-out fold was used for testing. Adam was used as the optimizer. We trained BrainGNN for 100 iterations with an initial learning rate of 0.001 and annealed to half every 20 epochs. Each batch contained 400 graphs for Biopoint data and 100 graphs for HCP data. The weight decay parameter was 0.005.

3 Results

3.1 Ablation Study and Hyperparameter Discussion

Ablation studies were performed to investigate the ROI-aware graph convolutional mechanism. We compared our proposed Ra-GNN layer with the strategy of directly learning embedding kernels W. We denoted the alternative strategy as 'GNN.' We tuned the coefficients $(\lambda_2^{(1)} - \lambda_2^{(2)} - \lambda_3)$ in the loss function in Eq. (11). $\lambda_2^{(1)}$ and $\lambda_2^{(2)}$ encouraged more separable node importance scores for selected and unselected nodes after pooling. λ_3 controlled the similarity of the nodes selected by different instances, which could control the level of interpretability between individual-level and group-level. Small λ_3 would result in variant individual-specific patterns, while large λ_3 would force the model to learn common group-level patterns. As task classification on HCP could achieve consistently high accuracy over the parameter variations, we only showed the results on the Biopoint dataset in Fig. 4 to better examine the effect of model variations.

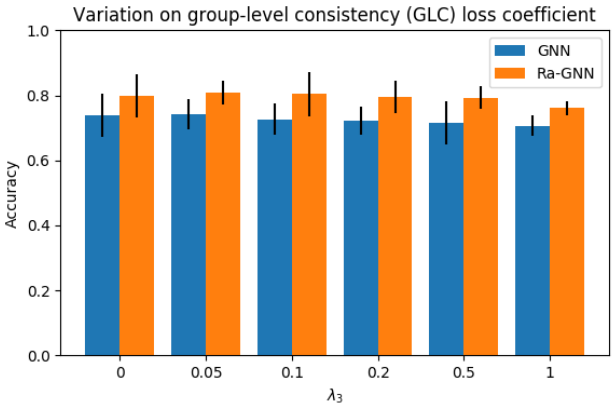
First, we investigated the effects of λ_3 on the accuracy, as a suitable range of λ_3 should be determined in order to not sacrifice model accuracy. In Fig. 4a, $\lambda_2^{(1)}$ and $\lambda_2^{(2)}$ were fixed to 0. We noticed that the results were stable to the variation of λ_3 in the range 0 - 0.5. When $\lambda_3 = 1$, the accuracy dropped. The

accuracy reached the peak when $\lambda_3 = 0.1$. As the other deep learning models behaved, BrainGNN was overparameterized. Without regularization ($\lambda_3 = 0$), the model was easier to overfit to the training set, while larger regularization on consistency might result in underfitting on the training set. Next, we fixed $\lambda_2^{(1)} = \lambda_2^{(2)} = 0.1$ and varied λ_3 again. As the results presented in Fig. 4b, the accuracy dropped if we increased λ_3 after 0.2, which followed the same trend in Fig. 4a. However, the accuracy under the setting of $\lambda_3 = 0$ was better than that in Fig. 4a. Probably the λ_2 terms worked as regularization and mitigated the overfitting issue. Then, we fixed $\lambda_3 = 0.1$ and varied $\lambda_2^{(1)}$ and $\lambda_2^{(2)}$ from $0 - 0.5$. As the results shown in Fig. 4c, when we increased $\lambda_2^{(1)}$ and $\lambda_2^{(2)}$ to 0.2, the accuracy slightly dropped, while the accuracy sharply dropped when they were increased to 0.5. For the following baseline comparison experiments, we set $\lambda_2^{(1)} - \lambda_2^{(2)} - \lambda_3$ to be $0.1 - 0.1 - 0.1$. As the results shown in Fig. 4, Ra-GNN overall outperformed the GNN strategy under all the parameter settings. The reason could be better node embedding from multiple embedding kernels in Ra-GNN, as the traditional GNN strategies treated ROIs (nodes) identically and used the same kernel for all the ROIs. Hence, we claim that Ra-GNN can better characterize the heterogeneous representations of brain ROIs.

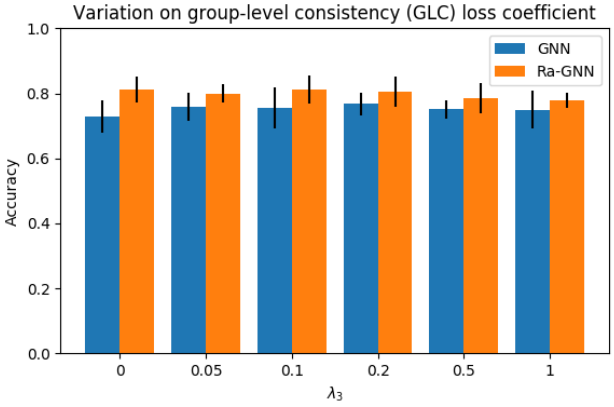
3.2 Comparison with Baseline Methods

First, we compared our method with traditional machine learning (ML) methods for fMRI analysis, which took vectorized correlation matrices as inputs. The ML baseline methods included Random Forest (1000 trees), SVM (RBF kernel), and MLP (2 layers with 20 hidden nodes). Second, we compared our method with the state-of-the-art deep learning (DL) methods, including BrainNetCNN [24], and other GNN methods: 1) replace Ra-GNN layer with the graph convolutional layers in Li et al. [28], 2) GraphSAGE [19] and 3) GAT (1 head) [43]. It is worth noting that GraphSAGE [19] and GAT [43] did not take edge weights in the aggregation step of the graph convolutional operation. The inputs of BrainNetCNN were correlation matrices. We used the parameter settings indicated in the original paper [24]. The inputs of the alternative GNN methods were the same as the inputs of BrainGNN and the hyper-parameter settings for the graphconv, pooling and MLP layers were the same as BrainGNN. The comparison results are shown in Fig. 5

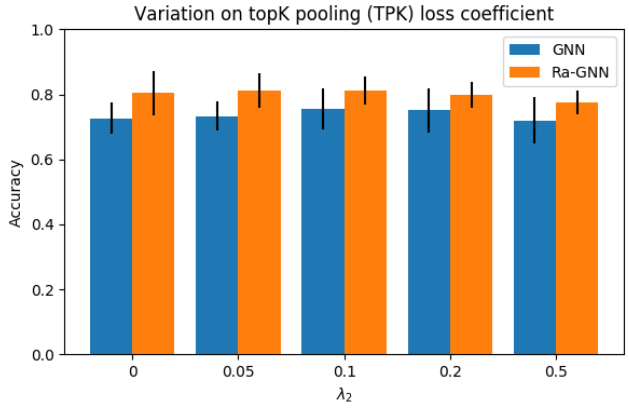
Our BrainGNN outperformed alternative models. The improvement may result from two causes. First, due to the intrinsic complexity of fMRI, complex models with more parameters are desired, which also explains why CNN and GNN-based methods were better than SVM and random forest. Second, our model utilized the properties of fMRI and community structure in the brain network and thus potentially modeled the local integration more effectively. Compared to alternative machine learning models, BrainGNN achieved significantly better classification results on two independent task-fMRI datasets. What is more, BrainGNN does not have the burden of feature selection, which is needed in traditional machine learning methods. Also, BrainGNN needs only $10 - 30\%$



(a) Variation on group-level consistency (GLC) loss coefficient λ_3, when setting $\lambda_2^{(1)} = \lambda_2^{(2)} = 0$

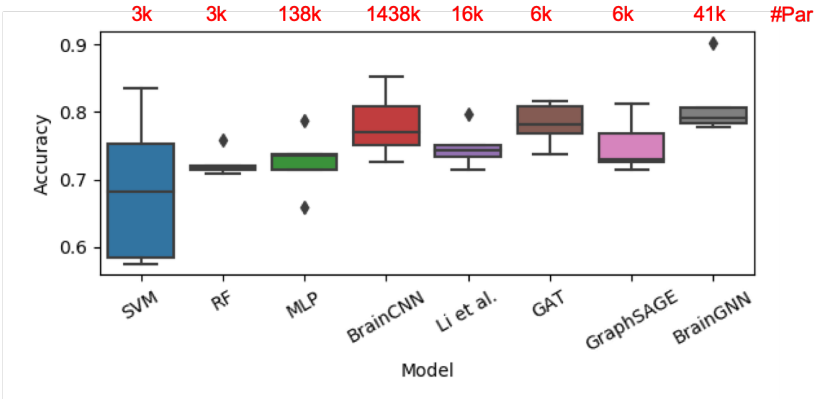


(b) Variation on group-level consistency (GLC) loss coefficient λ_3, when setting $\lambda_2^{(1)} = \lambda_2^{(2)} = 0.1$

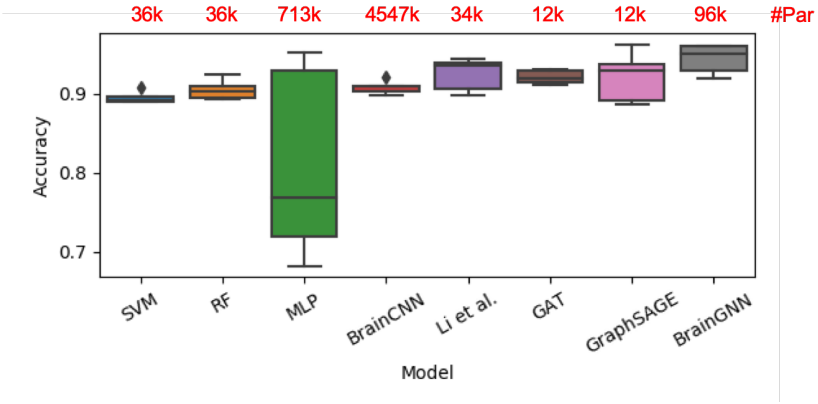


(c) Variation on topK pooling (TPK) loss coefficient $\lambda_2^{(1)}$ and $\lambda_2^{(2)}$ ($\lambda_2^{(1)} = \lambda_2^{(2)}$), when setting $\lambda_3 = 0.1$

Fig. 4: Ablation study comparison of ROI-aware GNN (Ra-GNN) and GNN without ROI embedding (GNN).



(a) Biopoint



(b) HCP

Fig. 5: Comparison of the classification accuracy of different baseline models. Classification accuracies of a 5-fold cross-validation study are depicted. The number of trainable parameters (♯Par) of the deep learning models are denoted on the top of each model in red.

of the number of parameters compared to MLP and less than 3% of the number of parameters compared to BrainNetCNN. Hence, BrainGNN is more suitable as a deep learning tool for fMRI analysis, as sample size is often limited.

3.3 Different Levels of Interpretation On Salient ROIs

Our proposed R-pool can prune the uninformative nodes and their connections from the brain graph based on the learning tasks. In other words, only the salient nodes would be kept/selected. We investigated how to control the level of interpretation by tuning the coefficient λ_3 that was associated with GLC loss. As we discussed in Section 2.4, large λ_3 led to group-level interpretation and small λ_3 led to individual-level interpretation. As we discussed in Section 3.1, when λ_3 is too large, the regularization might hurt the model accuracy. We put forth the hypothesis that reliable interpretation can only be guaranteed in terms of a model with high classification accuracy. Hence, the interpretation was restricted to models with fixed $\lambda_2^{(1)}$, $\lambda_2^{(2)}$ and varying λ_3 from 0 to 0.5 based on our experiments in Section 3.1. Without losing the generalizability, we showed the salient ROI detection results of 3 randomly selected ASD instances from the Biopoint dataset in Fig. 6. We showed the remaining 21 ROIs after the 2nd R-pool layer (with pooling ratio = 0.5, 25% nodes left) and corresponding pooling scores. As shown in Fig. 6(a), when $\lambda_3 = 0$, overlapped areas among the three instances were rarely to be found. In Fig. 6(b-c), we circled the big overlapped areas across the three instances. By visually examining the salient ROIs, we found three overlapped areas in Fig. 6(b) and five overlapped areas in Fig. 6(c). As proposed in Section 2.4, by tuning λ_3, BrainGNN could achieve different levels of interpretation.

3.4 Validating Salient ROIs

To summarize the salient ROIs over the five cross-validation folds, we averaged the node pooling scores after the 1st R-pool layer for all subjects across all folds per class. The top 20 salient ROIs were kept. We did not interpret the model from the 2nd R-pool layer as we did in Section 3.3, because the nodes left after the 1st R-pool layer may not be the same for different graphs. Therefore, it was infeasible to average the pooling scores without padding 0 scores to the unselected nodes. To validate the neurological significance of the result, we used Neurosynth [52], a platform for fMRI data analysis. Neurosynth collects thousands of neuroscience publications and provides meta-analysis, finding the keywords and their associated statistical images. The decoding function on the platform calculates the correlation between the input image and each functional keyword's meta-analysis images.

In Fig. 7(a-b), we displayed the salient ROIs associated with HC and ASD separately. Putamen, thalamus, temporal gyrus and insular, occipital lobe were selected for HC; frontal gyrus, temporal lobe, cingulate gyrus, occipital pole, and angular gyrus were selected for ASD. Hippocampus and temporal pole were important for both groups. The bar-chart in Fig. 7(c) illustrated the meta-analysis

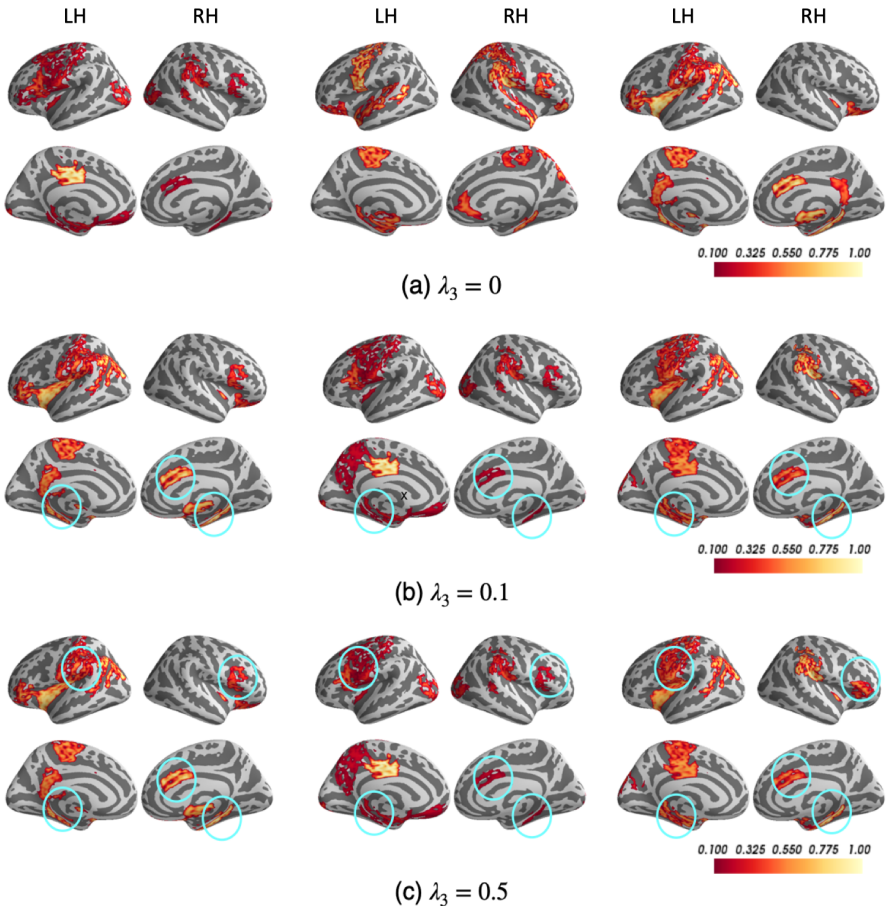
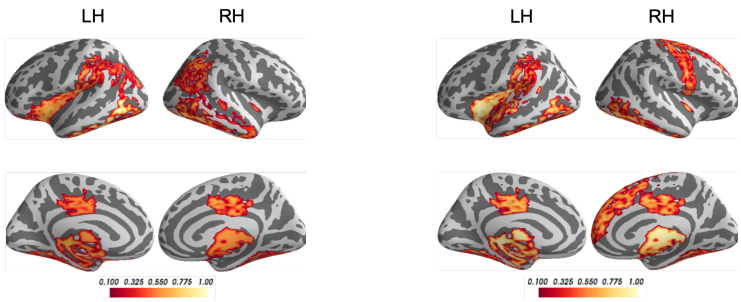
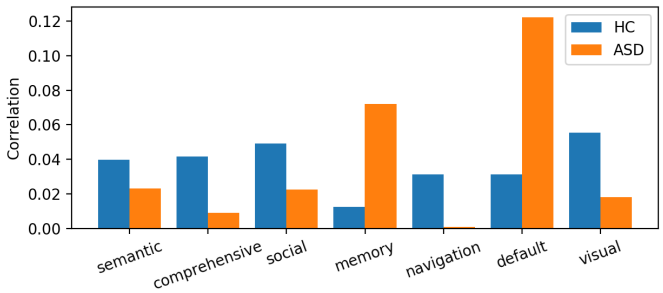


Fig. 6: The 21 selected salient ROIs of three different ASD individuals with different weights λ_3 associated with group-level consistency term L_{GLC}. The color bar ranges from 0.1 to 1. The bright-yellow color indicates a high score, while dark-red color indicates a low score. The common detected salient ROIs across different individuals are circled in blue.

on the functional keywords implied by the top 21 salient regions in HC and ASD groups using Neurosynth. We selected 'semantic', 'comprehension', 'social', 'memory', 'navigating', 'default' and 'visual' as the functional keywords, which were related to the Biopoint task [44]. We named the selected ROIs as the biomarkers for identifying each group. Recall that these topics reflected unbiased and aggregated findings across the fMRI literature. The functional dimensions in Fig. 7(c) exposed a clear functional distinction between the two groups in task fMRI decoding results. A higher value indicated a larger correlation to the functional keywords. Specifically, the biomarkers for HC corresponded to the areas of



(a) Salient ROIs associated with HC. (b) Salient ROIs associated with ASD



(c) Functional keywords decoding.

Fig. 7: Interpreting salient ROIs for classifying HC vs. ASD using BrainGNN (a-b) and decoding correlation scores of ROIs associated with the functional keywords using Neurosynth (c).

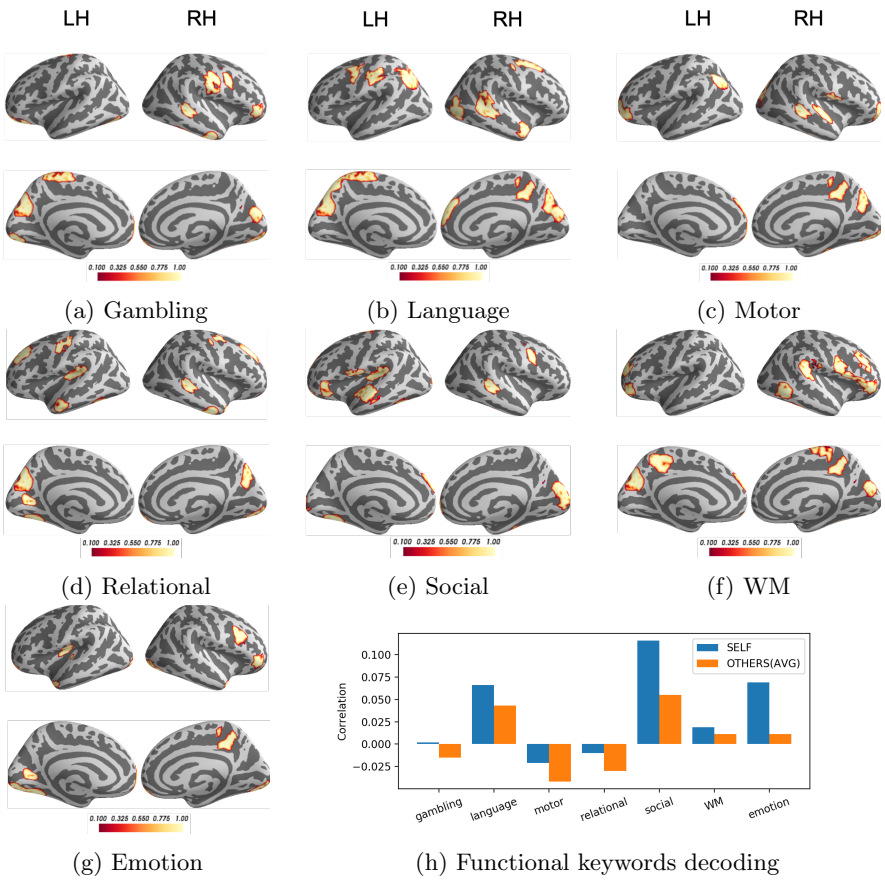


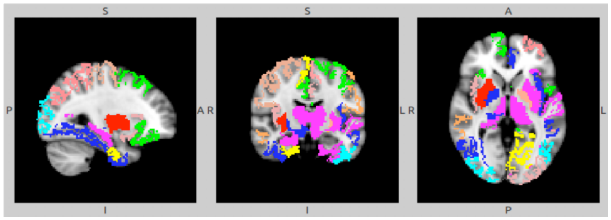
Fig. 8: Interpreting salient ROIs associated with classifying seven tasks (a-g) and decoding their correlation scores associated with the functional keywords using NeuroSynth (h). 'SELF' indicates the correlation score to the fMRI's real task category, and 'AVG(Others)' indicates the average of the scores to the other task categories.

clear deficit in ASD, such as social communication, perception, and execution. In contrast, the biomarkers of ASD mapped to implicated activation-exhibited areas in ASD: default mode network [5] and memory [4]. This conclusion is consistent both with behavioral observations when administering the fMRI paradigm and with a prevailing theory that ASD includes areas of cognitive strengths amidst the social deficits [37,41,21].

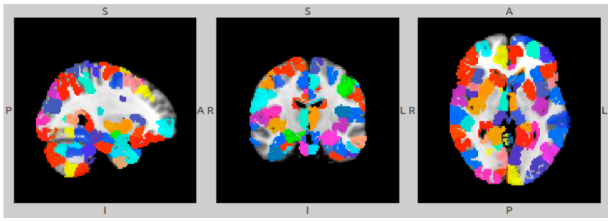
In Fig. 8(a-g), we listed the salient ROIs associated with the seven tasks for the HCP dataset. We selected 'gambling', 'language', 'motor', 'relational', 'social', 'working memory' (WM) and 'emotion' as the functional keywords, which

were exactly the functional design of the 7 tasks. The bar-chart in Fig. 8 (h) illustrated the meta-analysis on functional keywords implied by the top 21 salient regions corresponding to the seven tasks using Neurosynth. In all the seven tasks, salient ROIs corresponding to each task had higher Neurosynth score than the average of other tasks. The finding suggests that our algorithm identified ROIs that are key to distinguish between the 7 tasks. For example, the anterior temporal lobe and temporal parietal regions are selected for the social task, which are typically associated with social cognition [31,38]. Our findings also have overlaps with the task decoding results in recent works [47].

3.5 Node Clustering Patterns in Ra-GNN layer



(a) Biopoint



(b) HCP

Fig. 9: ROI clustering learned from θ_1 parameter from Ra-GNN layer. Different colors denote different communities.

We clustered all the ROIs based on the kernel parameter α_{ij}^{+} (learned in Eq. (3)) of the 1st Ra-GNN layer and showed the node clustering results for Biopoint and HCP data in Fig. 9a and Fig. 9b respectively. We used t-SNE [30] to visualize the raw node features of ASD in each community in Fig. 10a and their latent space embedded by the first Ra-GNN layer in Fig. 10b. The node representations in different communities were distinguishable, and the difference of node representations within the same community were magnified, which corroborated our assumption in Section 2.2 that different kernels accentuated diversified node representations in different communities. Similar patterns were observed in the HCP dataset.

(a) Node feature embedding for raw node representation $\mathbf{h}^{(0)}$.

(b) Node feature embedding for node representation $\mathbf{h}^{(1)}$ after the 1st Ra-GNN layer.

Fig. 10: t-SNE embedding for node features of ASD group in different communities, each color indicates a community. Before Ra-GNN layer, the node features are not distinguishable, while the nodes in different communities are more distinguishable after convolution by the Ra-GNN layer. Also, the spurting pattern in (b) demonstrates that the nodes in the same community have different representation that are more distinguishable from the other communities.

4 Discussion

In this paper, we propose a graph learning model, BrainGNN, for brain network analysis, that not only can perform prediction but also can be interpretable. We tested the algorithms on two datasets, Biopoint and HCP, to classify brain networks. Our main **contributions** are summarized as follows: 1) We formulate an end-to-end framework for fMRI prediction and biomarker interpretation. The methods can be generalized to general neuroimaging analysis; 2) We propose an ROI-aware GNN for brain graph node (ROI) embedding, which is parameter-efficient (Section 2.2) and interpretable for node clustering. Unlike other fMRI analysis methods that employ clustering as a preprocessing step to reorder nodes, BrainGNN learns the node grouping and extracts graph features jointly; 3) We modify topK pooling [15] for informative node selection and introduce a novel regularization term, topK pooling (TPK) loss (Section 2.3), to encourage more reasonable node selection; 4) By regulating intermediate outputs with a novel loss term, group-level consistency (GLC) loss, BrainGNN provides the flexibility to choose between individual-level and group-level explanations. To the best of our knowledge, we have not seen any previous research that provides flexible individual-level to group-level interpretation in GNN (Section 3.3).

4.1 Deep Learning Methods for fMRI Prediction

Deep learning is a promising data-driven tool to automatically learn complex feature representations in large data. Several deep learning approaches exist to

understand the human brain network [13,36,49]. A variety of deep methods have been applied to fMRI connectome data, such as the feedforward neural networks (FNN) [34], long short-term memory (LSTM) recurrent neural networks [9], and 2D convolutional neural networks (CNN) [24]. However, these existing deep learning methods for fMRI analysis usually require around millions of parameters to learn due to the high dimensionality of fMRI connectome, thus larger datasets are required to train the models. Compared with the above mentioned deep learning methods, GNNs require many fewer parameters and are designed for graph-structured data analysis. Hammond et al. [20] proposed a spectral graph convolution which defines convolution for graphs in the spectral domain. Later, Defferrard et al. [10] simplified spectral graph convolution to a local form and Kipf et al. introduced the Graph Convolutional Neural Network (GCN) [27] which provided an approximated fast computation. Hamilton et al. [19] proposed another variant of graph convolution in the spatial domain that improves GCN's scalability by using sampling-based neighborhood aggregation and applies GCN to inductive node embedding. Different from the GNN methods mentioned above, our proposed BrainGNN includes novel ROI-aware Ra-GNN layers that efficiently assign each ROI an unique kernel, revealing ROI community patterns and novel regulation terms (unit loss, GLC loss and TPK loss) for pooling operation that regulate the model to select salient ROIs. BrainGNN shows superior prediction accuracy for ASD classification and brain states decoding compared to the alternative machine learning, FCN, CNN and GNN methods. As it is shown in Fig 5), BrainGNN improves average accuracy between 3% and 20% for ASD classification on Biopoint dataset and achieves an average accuracy of 93.4% on a seven-class task states classification on HCP dataset.

4.2 Group-level and Individual-level Biomarker Analysis

Despite the high accuracy achieved by deep learning models, a natural question that arises is if the decision making process in deep learning models can be interpretable. One common property of linear regression and random forest is their interpretability of feature importance. For example, the coefficient of linear regression model and the Gini impurity gain associated with each feature in random forest can be seen as the importance scores of features. Data feature importance estimation is an important approach to understand both the model and the underlying properties of data. From the brain biomarker detection perspective, understanding salient ROIs associated with the prediction is an important approach to find the biomarkers, where the indicative ROIs could be candidate biomarkers.

Although deep learning model visualization techniques have been developed for convolution neural networks (CNNs), those methods are not directly applicable to explain weighted graphs with node features for the graph classification task. A few works [26,50,51] have discussed interpretable GNN models, where the internal model information such as weights or structural information can be accessed and inferred as group-level patterns for training instances only. Other

works have been used for explaining GNNs using post-hoc interpretation methods [35,3,53]. These post-hoc methods usually work by analyzing individual feature input and output pairs, which limits their explainability to individual-level only. Few GNN studies have explored both individual-level and group-level explanations, which are critical in neuroimaging research.

Here, we use model interpretability to address the issue of group-level and individual-level biomarker analysis. In contrast, without additional post-processing steps, the existing methods of fMRI analysis can only either perform individual-level or group-level functional biomarker detection. For example, general linear model (GLM), principal component analysis (PCA) and independent component analysis (ICA) are group-based analysis methods. Some deterministic models like connectome-based predictive modeling (CPM) [40,16] and other machine learning based methods provide individual level-analysis. However, the model flexibility for different-levels of biomarkers analysis might be required by different users. For precision medicine, individual-level biomarkers are desired for planning targeted treatment, whereas group-level biomarkers are essential for understanding the common characteristic patterns associated with the disease. To fill the gap between group-level and individual-level biomarker analysis, we introduce a tunable regularization term for our graph pooling function. By examining the pairs of inputs and intermediate outputs from the pooling layer, our method can switch freely between individual-level and group-level explanation under the control of the regularization term by end-to-end training. A large regularization parameter encourages interpreting common biomarkers for all the instances, while a small regularization parameter allows different interpretation for different instances.

We believe that BrainGNN is the first work that uses a single framework to transition between individual- and group-level analysis, filling the critical interpretation gap in fMRI analysis. The biomarker interpretation results can further help research in ASD and possibly generalize to rare diseases where there are few patients available, as it provides individual- to group-level biomarker associations.

4.3 BrainGNN as A Tool for Neuroimaging Analysis

Our proposed BrainGNN can be a research tool to identify autism biomarkers using whole-brain fMRI. Our proposed method will help support efforts to better understand the neural underpinnings of ASD, which is much needed in the field. A more precise understanding of the neural underpinnings will guide treatment approaches and help with the development of novel treatments, particularly innovative pharmacological interventions. It will also support the classification of subjects in research towards more homogeneous samples, which will increase power. The proposed method also provides researchers with the opportunity to study neural network decisions. The challenge in applying deep models to neuroimaging research is the black box feature of this approach: no one knows what the deep network is doing. Our proposed method is not only helpful for understanding the model mechanism, but also crucial for deciphering the human brain

network. The highly accurate results can furthermore help with classification and diagnosis of neuropsychiatric diseases [33].

4.4 Limitation and Future Work

The pre-processing procedure performed in Section 2.5 was one possible way of obtaining graphs from fMRI data, as demonstrated in this work. One meaningful next step is to use more powerful local feature extractors to summarize ROI information, such as embedding raw fMRI time series. A joint end-to-end training procedure that dynamically extracts graph node features from fMRI data is challenging, but an interesting direction. Also, in the current work, we only tried a single atlas for each dataset. In brain analysis, the reproducibility and consistency of the methods are important [48,1]. For ROI-based analysis, usually different atlases lead to different results [8]. It is worth further investigating whether the classification and interpretation results are robust to the choice of the atlas. Although we discussed a few variations of hyperparameters in Section 3.1, more variations should be studied, such as pooling ratio, the number of communities, the number of convolutional layers, and different readout operations. In future work, we will explore the connections between the Ra-GNN layer and the tensor decomposition-based clustering methods and the patterns of ROI selection and ROI clustering. For better understanding of the algorithm, we aim to work on quantitative evaluations and theoretical studies to explain the experimental results.

5 Conclusions

In this paper, we propose BrainGNN, an interpretable graph neural network for fMRI analysis. BrainGNN takes graphs built from neuroimages as inputs, and then outputs prediction results together with interpretation results. We applied BrainGNN on Biopoint and HCP fMRI datasets. With the built-in interpretability, BrainGNN not only performs better on prediction than alternative methods, but also detects salient brain regions associated with predictions and discovers brain community patterns. Overall, our model shows superiority over alternative graph learning and machine learning classification models. By investigating the selected ROIs after R-pool layers, our study reveals the salient ROIs to identify autistic disorders from healthy controls and decodes the salient ROIs associated with certain task stimuli. Certainly, our framework is generalizable to analysis of other neuroimaging modalities. The advantages are essential for developing precision medicine, understanding neurological disorders, and ultimately benefiting neuroimaging research.

References

1. Abraham, A., Milham, M.P., Di Martino, A., Craddock, R.C., Samaras, D., Thirion, B., Varoquaux, G.: Deriving reproducible biomarkers from multi-site resting-state data: an autism-based example. NeuroImage **147**, 736–745 (2017)
2. Baker, J.T., Holmes, A.J., Masters, G.A., Yeo, B.T., Krienen, F., Buckner, R.L., Öngür, D.: Disruption of cortical association networks in schizophrenia and psychotic bipolar disorder. JAMA psychiatry **71**(2), 109–118 (2014)
3. Baldassarre, F., Azizpour, H.: Explainability techniques for graph convolutional networks. arXiv preprint arXiv:1905.13686 (2019)
4. Boucher, J., Bowler, D.M.: Memory in autism. Citeseer (2008)
5. Buckner, R.L., Andrews-Hanna, J.R., Schacter, D.L.: The brain's default network: anatomy, function, and relevance to disease. (2008)
6. Cangea, C., et al.: Towards sparse hierarchical graph classifiers. arXiv preprint arXiv:1811.01287 (2018)
7. Carroll, J.D., Chang, J.J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of "eckart-young" decomposition. Psychometrika **35**(3), 283–319 (1970)
8. Dadi, K., Rahim, M., Abraham, A., Chyzhyk, D., Milham, M., Thirion, B., Varoquaux, G., Initiative, A.D.N., et al.: Benchmarking functional connectome-based predictive models for resting-state fmri. Neuroimage **192**, 115–134 (2019)
9. Dakka, J., Bashivan, P., Gheiratmand, M., Rish, I., Jha, S., Greiner, R.: Learning neural markers of schizophrenia disorder using recurrent neural networks. arXiv preprint arXiv:1712.00512 (2017)
10. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in neural information processing systems. pp. 3844–3852 (2016)
11. Desikan, R.S., Ségonne, F., Fischl, B., Quinn, B.T., Dickerson, B.C., Blacker, D., Buckner, R.L., Dale, A.M., Maguire, R.P., Hyman, B.T., et al.: An automated labeling system for subdividing the human cerebral cortex on mri scans into gyral based regions of interest. Neuroimage **31**(3), 968–980 (2006)
12. Du, Y., Fu, Z., Calhoun, V.D.: Classification and prediction of brain disorders using functional connectivity: promising but challenging. Frontiers in neuroscience **12**, 525 (2018)
13. Eickenberg, M., Varoquaux, G., Thirion, B., Gramfort, A.: Convolutional network layers map the function of the human visual cortex. ERCIM NEWS (108), 12–13 (2017)
14. Finn, E.S., Shen, X., Scheinost, D., Rosenberg, M.D., Huang, J., Chun, M.M., Papademetris, X., Constable, R.T.: Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity. Nature neuroscience **18**(11), 1664 (2015)
15. Gao, H., Ji, S.: Graph u-nets. arXiv preprint arXiv:1905.05178 (2019)
16. Gao, S., Greene, A.S., Constable, R.T., Scheinost, D.: Combining multiple connectomes improves predictive modeling of phenotypic measures. Neuroimage **201**, 116038 (2019)
17. Gong, L., Cheng, Q.: Exploiting edge features for graph neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9211–9219 (2019)
18. Greene, A.S., Gao, S., Scheinost, D., Constable, R.T.: Task-induced brain state manipulation improves prediction of individual traits. Nature communications **9**(1), 1–13 (2018)

19. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in neural information processing systems. pp. 1024–1034 (2017)
20. Hammond, D.K., Vandergheynst, P., Gribonval, R.: Wavelets on graphs via spectral graph theory. Applied and Computational Harmonic Analysis **30**(2), 129–150 (2011)
21. Iuculano, T., Rosenberg-Lee, M., Supekar, K., Lynch, C.J., Khouzam, A., Phillips, J., Uddin, L.Q., Menon, V.: Brain organization underlying superior mathematical abilities in children with autism. Biological Psychiatry **75**(3), 223–230 (2014)
22. Kaiser, M.D., Hudac, C.M., Shultz, S., Lee, S.M., Cheung, C., Berken, A.M., Deen, B., Pitskel, N.B., Sugrue, D.R., Voos, A.C., et al.: Neural signatures of autism. Proceedings of the National Academy of Sciences **107**(49), 21223–21228 (2010)
23. Karwowski, W., Vasheghani Farahani, F., Lighthall, N.: Application of graph theory for identifying connectivity patterns in human brain networks: a systematic review. frontiers in Neuroscience **13**, 585 (2019)
24. Kawahara, J., Brown, C.J., Miller, S.P., Booth, B.G., Chau, V., Grunau, R.E., Zwicker, J.G., Hamarneh, G.: Brainnetcnn: Convolutional neural networks for brain networks; towards predicting neurodevelopment. NeuroImage **146**, 1038–1049 (2017)
25. Kazi, A., Shekarforoush, S., Krishna, S.A., Burwinkel, H., Vivar, G., Kortüm, K., Ahmadi, S.A., Albarqouni, S., Navab, N.: Inceptiongcn: receptive field aware graph convolutional network for disease prediction. In: International Conference on Information Processing in Medical Imaging. pp. 73–85. Springer (2019)
26. Kim, B.H., Ye, J.C.: Understanding graph isomorphism network for brain mr functional connectivity analysis. arXiv preprint arXiv:2001.03690 (2020)
27. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
28. Li, X., Dvornek, N.C., Zhou, Y., Zhuang, J., Ventola, P., Duncan, J.S.: Graph neural network for interpreting task-fmri biomarkers. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 485–493. Springer (2019)
29. Loe, C.W., Jensen, H.J.: Comparison of communities detection algorithms for multiplex. Physica A: Statistical Mechanics and its Applications **431**, 29–45 (2015)
30. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
31. Mar, R.A.: The neural bases of social cognition and story comprehension. Annual review of psychology **62**, 103–134 (2011)
32. Moğultay, H., Alkan, S., Yarman-Vural, F.T.: Classification of fmri data by using clustering. In: 2015 23nd Signal Processing and Communications Applications Conference (SIU). pp. 2381–2383. IEEE (2015)
33. Nickerson, L.D.: Replication of resting state-task network correspondence and novel findings on brain network activation during task fmri in the human connectome project study. Scientific reports **8**(1), 1–12 (2018)
34. Patel, P., Aggarwal, P., Gupta, A.: Classification of schizophrenia versus normal subjects using deep learning. In: Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing. p. 28. ACM (2016)
35. Pope, P.E., Kolouri, S., Rostami, M., Martin, C.E., Hoffmann, H.: Explainability methods for graph convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10772–10781 (2019)

36. Rajalingham, R., Issa, E.B., Bashivan, P., Kar, K., Schmidt, K., DiCarlo, J.J.: Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. Journal of Neuroscience **38**(33), 7255–7269 (2018)
37. Robertson, C.E., Kravitz, D.J., Freyberg, J., Baron-Cohen, S., Baker, C.I.: Tunnel vision: sharper gradient of spatial attention in autism. Journal of Neuroscience **33**(16), 6776–6781 (2013)
38. Ross, L.A., Olson, I.R.: Social cognition and the anterior temporal lobes. Neuroimage **49**(4), 3452–3462 (2010)
39. Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: European Semantic Web Conference. pp. 593–607. Springer (2018)
40. Shen, X., Finn, E.S., Scheinost, D., Rosenberg, M.D., Chun, M.M., Papademetris, X., Constable, R.T.: Using connectome-based predictive modeling to predict individual behavior from brain connectivity. nature protocols **12**(3), 506 (2017)
41. Turkeltaub, P.E., Flowers, D.L., Verbalis, A., Miranda, M., Gareau, L., Eden, G.F.: The neural basis of hyperlexic reading: An fmri case study. Neuron **41**(1), 11–25 (2004)
42. Van Essen, D.C., Smith, S.M., Barch, D.M., Behrens, T.E., Yacoub, E., Ugurbil, K., Consortium, W.M.H., et al.: The wu-minn human connectome project: an overview. Neuroimage **80**, 62–79 (2013)
43. Veličković, P., et al.: Graph attention networks. In: ICLR (2018)
44. Venkataraman, A., Yang, D.Y.J., Pelphrey, K.A., Duncan, J.S.: Bayesian community detection in the space of group-level functional differences. IEEE transactions on medical imaging **35**(8), 1866–1882 (2016)
45. Von Luxburg, U.: A tutorial on spectral clustering. Statistics and computing **17**(4), 395–416 (2007)
46. Wang, J., Zuo, X., He, Y.: Graph-based network analysis of resting-state functional mri. Frontiers in systems neuroscience **4**, 16 (2010)
47. Wang, X., Liang, X., Jiang, Z., Nguchu, B.A., Zhou, Y., Wang, Y., Wang, H., Li, Y., Zhu, Y., Wu, F., et al.: Decoding and mapping task states of the human brain via deep learning. Human Brain Mapping (2019)
48. Wei, X., Warfield, S.K., Zou, K.H., Wu, Y., Li, X., Guimond, A., Mugler III, J.P., Benson, R.R., Wolfson, L., Weiner, H.L., et al.: Quantitative analysis of mri signal abnormalities of brain white matter with high reproducibility and accuracy. Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine **15**(2), 203–209 (2002)
49. Yamins, D.L., DiCarlo, J.J.: Using goal-driven deep learning models to understand sensory cortex. Nature neuroscience **19**(3), 356 (2016)
50. Yan, Y., Zhu, J., Duda, M., Solarz, E., Sripada, C., Koutra, D.: 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. pp. 772–782 (2019)
51. Yang, H., Li, X., Wu, Y., Li, S., Lu, S., Duncan, J.S., Gee, J.C., Gu, S.: Interpretable multimodality embedding of cerebral cortex using attention graph network for identifying bipolar disorder. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 799–807. Springer (2019)
52. Yarkoni, T., Poldrack, R.A., Nichols, T.E., Van Essen, D.C., Wager, T.D.: Large-scale automated synthesis of human functional neuroimaging data. Nature methods **8**(8), 665 (2011)

53. Ying, R., Bourgeois, D., You, J., Zitnik, M., Leskovec, J.: Gnn explainer: A tool for post-hoc explanation of graph neural networks. arXiv preprint arXiv:1903.03894 (2019)