ATTEND TO CONNECT: END-TO-END BRAIN FUNCTIONAL CONNECTIVITY ESTIMATION

Usman Mahmood, Zening Fu, Vince D. Calhoun, Sergey Plis
Tri-institutional Center for Translational Research in Neuroimaging and Data Science (TReNDS),
Georgia State University, Georgia Tech, Emory;
umahmood1@student.gsu.edu, {zfu,vcalhoun}@gsu.edu, s.m.plis@gmail.com

ABSTRACT

Functional connectivity (FC) studies have demonstrated the benefits of investigating the brain and its disorders through the undirected weighted graph of fMRI correlation matrix. Most of the work with the FC, however, depends on the way the connectivity is computed and further depends on the manual post-hoc analysis of the FC matrices. In this work, we propose a deep learning architecture (BrainGNN) that learns the connectivity structure while learning to classify subjects with schizophrenia. It simultaneously trains a graphical neural network on this graph and learns to select a sparse subset of brain regions important to the prediction task. We demonstrate the model’s state-of-the-art classification performance on a schizophrenia fMRI dataset and show how introspection leads to disorder-relevant findings. The graphs learned by the model exhibit strong class discrimination, and the identified sparse subset of relevant regions is consistent with the schizophrenia literature.

1 INTRODUCTION

Functional connectivity which is often computed using cross-correlation among brain regions of interest (ROIs) is a powerful approach which has been shown to be informative for classifying brain disorders and revealing putative bio-markers relevant to the underlying disorder (Liu et al., 2008; Lynall et al., 2010; Yu et al., 2017; Gadgil et al., 2020). Inferring and using functional connectivity through spatio-temporal data, e.g. functional magnetic resonance imaging (fMRI), has been an especially important area of research in recent times. Functional connectivity can improve our understanding of brain dynamics and improve classification accuracy for brain disorders e.g. schizophrenia (Yan et al., 2017; Parisot et al., 2018; Kawahara et al., 2016; Ktena et al., 2017).

Existing studies often heavily depend on the underlying method of functional connectivity estimation (Rashid et al., 2016; Saha et al., 2020; Salman et al., 2019). These studies work very well on classification but do not learn a sparse graph and not helpful for identifying bio-markers in the brain. Other studies Du et al. (2018) on brain disorders utilize ROIs predefined based on anatomical or functional atlases, which are either fixed for all subjects or are based on group differences. These approaches ignore the variations in ROIs due to the underlying disease conditions and discount the possibility that only a small subset of ROIs may be important at a time.

In this work, we address the problems of using a fixed method of learning functional connectivity and using it as a fixed graph to represent brain structure (the standard practices) by utilizing a novel attention based Graph Neural Network (GNN) Li et al. (2016), which we call BrainGNN. We apply it to fMRI data and 1) achieve comparable classification accuracy to existing algorithms, 2) learn dynamic graph functional connectivity, and 3) increase model interpretability by learning which regions from the set of ROIs are relevant for classification.
Figure 1: BrainGNN architecture using a) Preprocessing: To preprocess the raw data with different steps (2.1.1), b) 1DCNN: To create embedding for regions (2.2.1), c) Self-attention: To create connectivity between regions (2.2.2) d) GNN: To obtain a single feature vector for the entire graph (2.2.3) and e) Linear classifier: To obtain the final classification.

2 MATERIALS AND METHODS

2.1 MATERIALS

In this study, we worked with the data from Function Biomedical Informatics Research Network (FBIRN) Keator et al. (2016) dataset including schizophrenia (SZ) patients and healthy controls (HC) for testing our model. Resting fMRI data from the phase III FBIRN were analyzed for this project.

2.1.1 PREPROCESSING

The FBIRN dataset was pre-processed through SPM12 Penny et al. (2011) based on the MATLAB 2019 environment. The slice-timing was first performed on the data, and then subject head motions were corrected by the realignment procedure. After that, the data was warped to MNI space using EPI template and resampled to 3 mm³ voxels. Finally, the data were smoothed with a 6mm FWHM Gaussian kernel. We selected subjects for further analysis Fu et al. (2021a) if the subjects have head motion ≤ 3° and ≤ 3 mm, and with functional data providing near full brain successful normalization (Fu et al., 2019). This resulted in a total of 311 subjects with 151 healthy controls and 160 subjects with schizophrenia. Each subject is represented by $X \in \mathbb{R}^{x \times y \times z \times t}$, where $x, y, z$ represent the number of voxels in each dimension and $t$ is the number of time points which are 160. To reduce the affect of noise we z-score the time sequence of each voxel independently. To partition the data into regions use automated anatomical labeling (AAL) Tzourio-Mazoyer et al. (2002) which contains 116 brain regions. We take the weighted average of the voxel intensities inside a region. This results in a dataset $D = (S_1, S_2, S_3, \ldots , S_n)$ where $S_i \in \mathbb{R}^{r \times t}$, $n = 311$, $r = 116$, $t = 160$.

2.2 METHOD

We have three distinct parts in our novel attention based GNN architecture: 1) a Convolutional Neural Network (CNN) Lecun et al. (1998), 2) a Self-Attention mechanism Vaswani et al. (2017), and 3) a GNN (Graph Neural Network). In this section we explain the purpose and details of each part separately. Refer to Figure 1 for the complete architecture diagram of BrainGNN.

2.2.1 CNN ENCODER

We use a CNN Lecun et al. (1998); Kiranyaz et al. (2021) encoder to obtain the representation of individual regions created in the preprocessing step outlined in 2.1.1. Each region vector of dimension $t = 160$ is passed through multiple layers of one dimensional convolution, and a fully connected layer to get final embedding. We use rectified linear unit (ReLU) as an activation layer between
convolution layers. Our one dimensional CNN layer embeds the temporal features of regions and
the spatial connections are handled in the attention and GNN parts of the architecture.

2.2.2 SELF ATTENTION

Using the embeddings created by the CNN encoder, we estimate the connectivity between brain
regions using multi-head self-attention following (Vaswani et al., 2017). The self-attention model
creates three embeddings namely (key, query, value) for each region, using three simple linear layers.
To create weights among regions, the model takes dot product of a region’s query with every other
region’s key embedding to get scores between them. Hence, \( \text{score}_{ij} = \text{query}_i \cdot \text{key}_j \). The scores
are then converted to weights using softmax, \( w_i = \text{Softmax}(\text{score}_i) \). Weights are multiplied with
each region’s value embedding and summed together to create new representation for \( \text{region}_i \).

This process is carried out for all regions, producing new representation of every region and weights
between regions. These weights are then used as functional connectivity between brain regions. The
self attention layer encodes the spatial axis for each subject and provides with connection between
regions. Weights are learned via end to end learning of our model performing classification. This
frees us from using predefined models or functions to estimate the connectivity.

2.2.3 GNN

Our graph network is based on a previously published model Li et al. (2016). Each subject is rep-
resented by a graph \( G \) having \( V,A,E \) where \( V \in \mathbb{R}^{r \times t} \) is the matrix of vertices, where each vertex
is represented by an embedding acquired by self-attention. \( A,E \in \mathbb{R}^{r \times r} \) are the adjacency and
edge weight matrices. Since we do not use any existing method of computing edges, we construct
a complete directed graph with backward edges, meaning every pair of vertices is joined by two
directed edges with weights \( e_{ij} \) and \( e_{ji} \in E \). For each GNN layer, at every step \( s \), each node/region,
sums feature vectors of every other region relative to the weight edge between the nodes and pass
the resultant and it’s own feature vector through a gated recurrent unit (GRU) network Cho et al.
(2014), to obtain new embedding for itself. \( x^n_i = \text{GRU}(x^n_{i-1}, \sum_{j:n_j > n_i} e_{ji} x^n_j) \).

The number of steps is a hyper-parameter which we have set it as 2 based on our experiments.
The graph neural network helps nodes to create new embeddings based on the embeddings of other
regions in the graph weighted by the edge weights between them. We use 6 GNN layers, with the first
3 followed by a top-k pooling layer Gao & Ji (2019); Knyazev et al. (2019). Pooling is performed to
help model focus on the important regions/nodes which are responsible for classification. Since we
represent each subject as graph \( G \), in the end we do graph classification by pooling all the feature
vectors of the remaining 23 regions/nodes. To get one feature vector from the entire graph we
concatenate the output of three different pooling layers. We use graph max pool, graph average pool
and attention based pool Vinyals et al. (2016). The resultant feature vector is then passed through a
single linear layer for classification.

2.2.4 TRAINING AND TESTING

To train, validate and test our model we divide the total 311 subjects into three groups of size 215,
80 and 16, for training, validating and testing respectively. We use 19 fold cross validation with 10
trials per fold, resulting in a total of 190 trials. We randomly select 100 subjects per class for each
trial. We calculate the area under the ROC (receiver operating characteristic) curve (AUC) for each
trial. To optimize our model we train all of our architecture in an end to end fashion, using Cross
Entropy to calculate our loss by giving true labels \( Y \) as targets. Let \( \theta \) represent the parameters of the
entire architecture. \( \text{loss} = \text{CrossEntropy}(\hat{Y}, Y) ; \theta^* = \arg \min_{\theta} (\text{loss}; \theta) \)

3 RESULTS

We show three different groups of results in our study. 1) The classification results, 2) Regions’
connectivity and 3) Key regions selection. We discuss these in the following sections. The input for
the machine learning models is sFNC matrices produced using Pearson product-moment correlation
coefficients (PCC).
a) Classification

We use AUC metric to quantify the classification (SZ/HC) results of our model. The performance is comparable to state of the art classical machine learning algorithms using hand crafted features and existing deep learning approaches such as Mahmood et al. (2020), which performed experiments on independent component analysis (ICA) components. Figure 2 shows that BrainGNN provide state of the art results. BrainGNN provides mean AUC as 0.79, which is just (∼ 0.02) less than the best performing model (SVM).

b) Functional Connectivity

Figure 3a shows weight matrices for the second test set in cross validation. Weight matrices of subjects belonging to SZ class turn out to be much sparser than weights of healthy controls subjects. The result shows that the connectivity is limited to fewer regions, and functional connectivity differs across classes and fewer regions get higher weights in case of SZ subjects. FNC matrices produced using PCC method 3b, do not provide such level of information and almost all regions get unit weight between other regions.

c) Region Selection

As hypothesized, Figure 4a shows that not all regions are equally important/informative for the underlying disease. Figure 4a shows the final 23 regions selected after the last pooling layer in the GNN model (using 2nd test fold) which is just 20 percent of the total brain regions used. Figure 4b shows the location of the selected regions in the MNI brain space, regions are distinguished by color.
Figure 4: 4a: Regions selected after the last pooling layer of GNN. All 23 regions are selected equal number of times (16). 4b: Mapping the 23 regions back on the brain across the three anatomical planes. 101th time point is selected for these brain scans. X axis shows different slices of the plane.

4 CONCLUSION

High classification (SZ/HC) performance shows that the model can accurately classify the subjects and hence can be trusted with the other two interpretative results of the paper. Functional connectivity between regions shown in the paper is of paramount importance as it highlights how brain regions are connected to each other and the variation between classes. The final regions selected by the model strengthens our hypotheses that not all regions are equally important for identifying a particular brain disorder. The regions selected by our model such as (cerebellum, temporal lobe, caudate, SMA) etc have been linked to the disease by multiple previous studies, hence reassuring the correctness of our model Jones et al. (2012); Fu et al. (2021b); Ebdrup et al. (2010); Andreasen & Pierson (2008). BrainGNN almost eliminates manual decisions transitioning graph construction and region selection into the data-driven realm. We envision in a future extension of the model that would enable it to work directly from the voxel-level not only connecting and selecting ROIs, but also constructing them.

5 ACKNOWLEDGMENTS

This work was in part supported by NIH grants 2R01EB006841 and 1RF1MH121885. Data for Schizophrenia classification was used in this study were downloaded from the Function BIRN Data Repository (http://bdr.birncommunity.org:8080/BDR/), supported by grants to the Function BIRN (U24-RR021992) Testbed funded by the National Center for Research Resources at the National Institutes of Health, U.S.A. and from the COLlaborative Informatics and Neuroimaging Suite Data Exchange tool (COINS: http://coins.trendscenter.org) and data collection was performed at the Mind Research Network, and funded by a Center of Biomedical Research Excellence (COBRE) grant 5P20RR021938/P20GM103472 from the NIH to Dr. Vince Calhoun.

REFERENCES


