# FMRI analysis using Graph Convolutional Neural Network

Manishkumar Alagumalai
*MS in Intelligent Systems Engineering*
*Maalag@iu.edu*

Rishit Puri
*MS in Data Science*
*rispuri@iu.edu*

Aathirai Senthilkumar Thamaraiselvi
*MS in Data Science*
*aasent@iu.edu*

Aravind Sivakumar
*MS in Computer Science)*
*arsivak@iu.edu*

Aaditya Dharne
*MS in Data Science*
*aadharne@iu.edu*

***Abstract*—The project aims to study the temporal statistical relationship among different regions of the human brain and draw meaningful insights using that information. We can use this information to understand the entity in a number of ways, such as determining his IQ, the activity he is engaged in, if the entity is a child or an adult, etc. The relationship information from brain activity is obtained by the use of FMRI (Functional Magnetic Resonance Imaging) technology. The brain regions that coactivate are represented by nodes, and the strength of the activations between those regions is represented by edges, in matrices or graphs that we generate using this information. The primary aim of this study is to utilise the FMRI data to determine the age of the participant performing a particular activity. We can conclude that this is a classification issue based on the available information.**

***Index Terms*—FMRI, GNN, OpenNeuro, GCN, EdgeConv, GAT, GraphSage, Graph Neural ODE.**

## I. Introduction

FMRI (Functional Magnetic Resonance Mapping) is a neuroimaging technique that measures brain activity by detecting changes in blood flow. It works similarly to an MRI scan, but instead of examining organs, fMRI looks at the function of the brain and is often used in brain surgery. This project is based on GNN, and we know that the brain structure is closely associated with GNN. The brain can be conceptualized as a complex network where each brain region is a node and connections (edges) between regions represent functional interactions. By leveraging the MRI data of 3–12-year-old children and adults during the viewing of a short animated film, Pixar "Partly Cloudy", our goal is to predict whether a participant watching the film is an adult or a child. This is a binary prediction problem where the brain's functional connectivity patterns are used as characteristics for classification. Essentially, this study offers a fascinating intersection of developmental psychology, neuroscience, and machine learning, providing fresh perspectives on the dynamic nature of the human brain.

## II. Related Work

Historically, graph-based fMRI analysis has utilized a two-stage methodology. The first stage involves deriving features from graphs via graph theoretical metrics to transform connectivity data into quantifiable statistics, followed by the analysis of these features. Given the complex and voluminous nature of fMRI data, regions of interest (ROIs) are typically grouped into densely linked communities to streamline data structures or enhance feature selection based on the data. However, inaccuracies in the initial feature extraction phase can lead to substantial errors in the subsequent analytical phase.

In recent times, graph neural networks (GNNs) have emerged as a leading technique for managing graph-structured data. These networks merge node and edge features with the overall graph configuration, resembling an adaptation of convolutional neural networks used for image processing. This synthesis enables a more efficient and interpretable analysis of graph data. Modern GNN methodologies acknowledge the unique identities and positions of each node within the brain's graph layout, a critical factor that permits a more detailed and individualized analysis of neuroimaging data. The paper X. Li et al., unveil a novel GNN framework that utilizes these insights to delineate both regional and cross-regional functional activation patterns for specific classification tasks, such as distinguishing neuro-disorder patients from healthy controls and decoding cognitive tasks. This approach addresses previous methodological oversights by recognizing each brain ROI as unique within the graph and implementing a clustering-based method for node embedding in the GNN's convolutional layers. This technique reduces potential biases from earlier models and enhances the understanding of clustering patterns pertinent to other brain imaging analyses. Moreover, the model offers greater interpretability through innovative loss terms that manage the aggregation of graph nodes, facilitating analyses that range from individual to group-level insights.

This expanded paper builds upon their initial conference presentation by introducing sophisticated graph convolutional layers and applying these methods to a new dataset and task, advancing beyond their preliminary work presented at the

International Conference on Medical Image Computing and Computer-Assisted Intervention. The BrainGNN framework incorporates unique Ra-GConv layers that assign specific kernels to each ROI, reflecting their community patterns, and employs advanced regularization techniques (unit loss, GLC loss, and TPK loss) to improve the selection process of essential ROIs. This approach has shown superior performance in classifying Autism Spectrum Disorder (ASD) and decoding various brain states, surpassing traditional machine learning methods like MLP, CNN, and other GNN techniques.

In this research, the authors introduce BrainGNN, a graph neural network that interprets fMRI data. This network processes graphs derived from neuroimages, yielding both predictive and interpretative outcomes. BrainGNN was evaluated using the Biopoint and Human Connectome Project (HCP) fMRI datasets. Due to its built-in interpretability, BrainGNN not only produces better predictive outcomes than other models but also successfully identifies significant brain regions and clarifies patterns within brain communities. The findings underscore the model's superiority over traditional graph learning and machine learning classification methods. By examining ROIs post-R-pool layers, the study pinpointed key regions essential for differentiating between autistic and healthy individuals and for decoding brain activities tied to specific tasks. This capability showcases the potential of their model to be applied across various neuroimaging techniques, substantially benefiting precision medicine, advancing the understanding of neurological disorders, and promoting neuroimaging research.

In the study detailed in the paper Lebo Wang, Kaiming Li, Xiaoping P. Hu, the CGCN architecture was introduced for fMRI analysis and applied to two distinct classification tasks using resting-state fMRI data: individual identification and Autism Spectrum Disorder (ASD) classification. The CGCN outperformed previous models by effectively capturing the spatial characteristics of fMRI data among connectomic neighbors.

Instead of utilizing hard thresholding to determine functional connectivity, a k-NN graph approach was adopted, which reduces the size of the network neighborhood and lessens noise impacts in connectivity, as evidenced by research from Liu, Nalci, and Falahpour (2017); and Murphy and Fox (2017). The k-NN graph method has several advantages, notably ensuring natural local uniformity with an equal number of connections emanating from each node, ideal for convolution operations. It also supports hierarchical feature extraction through the layer stacking enabled by the k-NN graph's structure.

In the paper Richardson, H., Lisandrelli, G., Riobueno-Naylor, A. et al., it was discussed that significant cognitive and brain development occurs in children's early years, especially in social cognition. Before starting formal education at six years old, children develop a complex understanding of others' desires, thoughts, and emotions, separate from physical sensations like pain or illness. While brain regions associated with Theory of Mind (ToM) have been extensively studied in adults, adolescents, and older children, conducting fMRI studies on very young children presents significant challenges. However, using the engaging and brief Pixar movie "Partly Cloudy," which depicts the physical sensations and mental states of its characters—a cloud named Gus and his stork friend Peck—we successfully collected functional data from a large group of children, including 65 between the ages of 3 and 6. This method allowed us to observe the development of cortical networks involved in processing bodily sensations and mental states, linking these changes to behavioral developments in ToM.

The findings indicate that while watching an animated movie that depicts the internal states of its characters, children activate specific cortical networks for processing states of the mind versus those of the body. These networks are distinct even in 3-year-old children and exhibit greater internal coherence and reduced cross-network interaction as they age. The degree of anti-correlation between these networks serves as a strong indicator of their developmental maturity in response to the movie. Moreover, certain key scenes in the movie trigger brain activity that increases with age and ToM reasoning capability. Conversely, achieving well-known ToM behavioral milestones, such as passing explicit false-belief tasks, does not seem to correspond with distinct changes in the neural mechanisms that underpin understanding the minds of others.

## III. Method

In this part, we'll look at how to use GNNs to classify subjects based on fMRI information. Our method can be broken down into two parts: data preparation and graph preparation, followed by GNN model structure and training.

### A. Data Preprocessing and Graph Construction

The first step is to get fMRI data on a sample of subjects. To do this, we decided to use a public ADHD dataset, which could have been downloaded from the learn website. This dataset provides fMRI data that has been preprocessed on a sample of individuals (either adults or children). The masker function is used during preprocessing to make sure that the data is consistent across subjects and across brain regions (Voxels). The masker standardizes time series data and normalizes the signal intensity variation.

The next step is to calculate correlation matrices, which measure the functional connectivity of brain regions. A correlation matrix is a representation of the degree of synchronization between the activity patterns of two brain regions. For indirect correlations, a partial correlation matrix can be used to account for the relationships observed. Once we have extracted the appropriate features, we create a dataset that fits our GNN models. To do this, we define a customized dataset class, which inherits from pytorch geometric's in-memory dataset. This class makes it easy to load and preprocess data for GNN models.

A function is executed within the dataset class to process the correlation / partial correlation matrices of each subject. The function determines the top neighbors of each subject. These top neighbors are the most correlated regions of interest

(ROIs) of each subject. In this selection process, the function creates a sparse adjacency matrix that encodes the structure of a functional connectivity network. In a GNN model, the correlation matrix itself is the node feature. The correlation matrix encapsulates the pairwise relationship between the ROIs. This information is useful to the model for learning. Subject labels (adult/child) are extracted from a different CSV file and transformed into pytorch tensors in a format that is easily understood by the model.

### *B. GNN Model Architecture*

- Our research examines the effectiveness of three different GNN architectures in fMRI classifying fMRI: graph convolutional networks (GCN), graph attention networks (GAT), Edge Conv networks, and graph sage networks (GANs) [**Fig 1**]. These models work on graphs, with nodes representing brain regions, and edges representing the functional connections between those regions. In the first architecture, a GCN, we use GNCConv layers to fine-tune the node representations. These layers aggregate information from neighboring regions of interest (ROIs) and weight it based on the correlation strengths we capture in our adjacency matrices. This helps the GCN learn more about each brain region's activity relative to its interconnected network.
- The second architecture is the GAT. GATs use the attention mechanism. Unlike GCN, which treats all neighbors the same, the attention mechanism allows GATs to selectively attend to relevant neighbors. By doing so, it is possible for GATs to achieve better-focused learning and better classification performance.
- The third architecture is the GraphSage. The GAT uses the GATconv layers. The GATconv layer can be a variant of the GCNConv layer or a variant of the GATConve layer. GraphSage uses the message-passing scheme. Each node aggregates the information from the neighbors using a pre-defined function. This helps GraphSage learn to represent informative node representations in graphs with sparse historical data.
- While the specific details of the GraphSage implementation will be elaborated upon later, all three GNN models share a similar overall architecture:
  Input Layer: This is where the node features (Correlation matrix) are received for each return on investment (ROI). Convolutional Layers: The model stacks several convolutional layers, each of which fine-tunes the node representations by including information from neighboring returns on investments (ROIs). Global Pooling Layer: Aggregates information from all the nodes on the graph. This aggregation can be done by summation or averaging. Output Layer: The graph-level representation is taken as input for the fully connected layer with the softmax activation function. Output Layer Class Probability (adult/child): The softmax function is used to classify the subject based on its fMRI data.

```
Input (x)
   |
MLP1 (Linear + ReLU)
   |
EdgeConv1 (Max Aggr)
   |
BatchNorm1
   |
ReLU
   |
MLP2 (Linear + ReLU)
   |
EdgeConv2 (Max Aggr)
   |
BatchNorm2
   |
ReLU
   |
MLP3 (Linear + ReLU)
   |
EdgeConv3 (Max Aggr)
   |
BatchNorm3
   |
Global Mean Pooling
   |
Linear (Output)
   |
Softmax (Classification)
Output
```

Fig. 1. Base GNN Network Architecture.

## IV. EXPERIMENT SETUP

### *A. Data Analysis*

For this research, the dataset was sourced from the **OpenNeuro** repository, a publicly accessible database hosting neuroimaging datasets. We utilized the **Nilearn** Python package to interface with the OpenNeuro platform directly, facilitating seamless access to the preprocessed dataset. The dataset underwent preprocessing using **fMRIPrep**, a widely utilized tool specifically designed for the preprocessing of functional magnetic resonance imaging (fMRI) data. fMRIPrep ensures standardized preprocessing steps, including motion correction, spatial normalization, and artifact removal, thereby enhancing the reliability and comparability of the dataset. The study used additional packages like **Matplot** in combination with Nelearn connectome packages to properly understand the underlying dataset, finding the relationship by drawing correlation plots[**figure 3 and figure 4**] between different regions of the brain.

The dataset comprises **144 records** of brain signal data, encompassing both child and adult participants. Each participant's brain signals were recorded continuously for a duration of 5 minutes while they watched a Pixar movie stimulus. The dataset captures neural activity during the viewing experience, providing insights into the brain's response to dynamic visual stimuli. The inclusion of both child and adult participants allows for the exploration of developmental differences in neural processing during cinematic engagement.

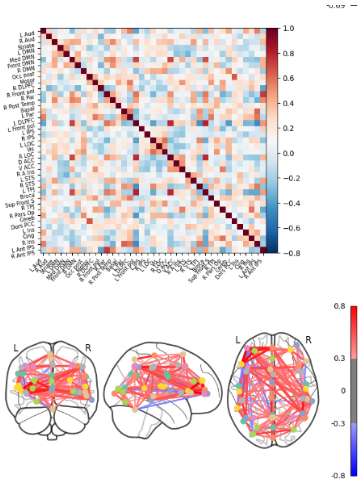

Fig. 2. Child's Brain

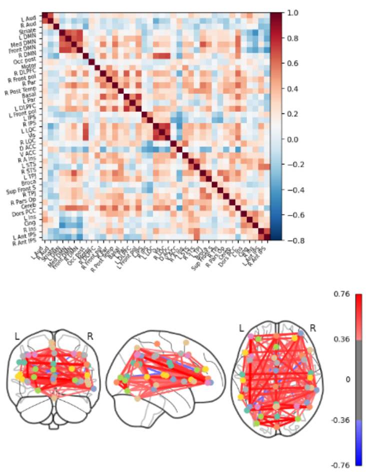

Fig. 3. Adult's Brain

Fig. 4. Correlation plots

### B. Instance Setup

The research was conducted entirely using the **Google Collab T4 GPU** instance. The instance comes equipped with **12 GB** of VRAM which is suitable for running GPU-intensive tasks like training an image classification network, small LLM models, etc.

### C. Dataset and Hyperparameter

After assessing the interrelationships within the dataset using the correlation matrix, the study leveraged partial correlation weights to construct a structured dataset. The **scikit-learn (sklearn)** package facilitated the shuffling of the dataset and its division into distinct training and testing subsets, ensuring robust model evaluation.

To represent the dataset in a format amenable to graph-based analyses, the **NetworkX** library's **from-numpy-matrix** function was employed. This transformation facilitated the conversion of the dataset into a graph structure suitable for processing by Graph Neural Networks (GNNs), enabling the extraction of meaningful insights from the interconnected data.

The GNN architecture comprised multiple convolutional layers, including EdgeConv, SageConv, GATConv, and GCNConv, designed to capture intricate patterns within the graph structure. A final dense layer, equipped with a softmax activation function, was incorporated to predict the probability of a given sample belonging to the Child or Adult class.

Throughout model training, the rectified linear unit (**ReLU**) activation function was utilized to introduce non-linearity, enhancing the network's expressive power. The training process spanned **30** epochs, with a batch size of 32 samples and a learning rate set to **0.001**, ensuring sufficient convergence and model stability.

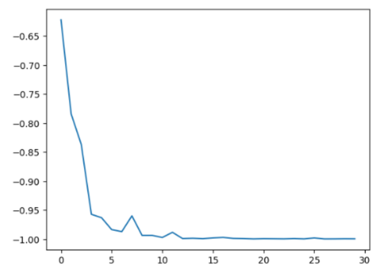


Fig. 5. Training Loss plot for GCN network

## V. Results

The research investigation encompassed a comprehensive exploration of multiple Graph Neural Network (GNN) architectures. Practical experimentation revealed noteworthy performance across various GNN models, notably Graph Sage and Graph ODE, as illustrated in [**Figure 6**]. Initial experimentation focused on the Graph Convolutional Network (GCN) architecture. Remarkably, the GCN network demonstrated exceptional capability in learning features from edge relationships, resulting in a substantial reduction in overall loss, as depicted in [**Figure 5**].

**Comparison table**

| Model | No of epochs trained | Test accuracy(%) |
| --- | --- | --- |
| GCN Conv Layers | 30 | 77.42 |
| EDGE Conv Layers | 30 | 87.10 |
| GAT | 30 | 93.55 |
| Graph Neural ODE | 30 | 93.55 |
| Graph Sage | 30 | 96.77 |

Fig. 6. Architecture Comparison

The experimentation process entailed the systematic exploration of different hyperparameters, including learning rates and batch sizes. Intriguingly, the network exhibited optimal performance at lower learning rates, suggesting the importance of fine-tuning hyperparameters for optimal model convergence and performance enhancement.

## VI. Conclusion

This research shed some light on how Neuro data along with the power of machine learning, and graph analysis can be used to predict what a participant is doing, what emotions the participant is feeling, or what is the participant's age in our case. This research was also helpful in understanding different packages like Nilearn, NetworkX can be used to understand and convert brain signal data into graph matrices which can make the GNN network training easier. Moving forward, further investigations into advanced GNN architectures to unlock new insights into the intricacies of human brain function and behavior.

Since the total number of training and test examples was small, there was a clear overfitting on the training dataset. This could be fixed by adding more training and validation samples. Furthermore, the research could be improved by adding more labels by using different age groups and checking if the network does better on multiple class classification. This research can be used as a stepping stone to create a better and more reliable classifier that can be able to predict age, emotions, etc

## VII. Team Member Contribution

**Manishkumar Alagumalai** - Report Writing, Training, and testing multiple Networks on the dataset, Codebase setup, and initial data analysis.

**Rishit Puri** - Report writing, Model Building(Graph Sage), Data analysis, Debugging.

**Aaditya Dharne** - Report Writing, Presentation help.

**Aathirai Senthilkumar Thamaraiselvi** - Initial data analysis, presentation preparation.

**Aravind Sivakumar** - Developing models, briefly reviewing research papers and contemplating the integration of newer models such as GraphODE.

## References


[1] X. Li et al., "BrainGNN: Interpretable Brain Graph Neural Network for fMRI Analysis," Medical Image Analysis, vol. 74, p. 102233, Dec. 2021, doi: https://doi.org/10.1016/j.media.2021.102233.

[2] H. Zhang et al., "Classification of Brain Disorders in rs-fMRI via Local-to-Global Graph Neural Networks," IEEE Transactions on medical imaging, vol. 42, no. 2, pp. 444–455, Feb. 2023, doi: https://doi.org/10.1109/tmi.2022.3219260.

[3] T. Azevedo et al., "A deep graph neural network architecture for modeling spatio-temporal dynamics in resting-state functional MRI data," Medical Image Analysis, vol. 79, p. 102471, Jul. 2022, doi: https://doi.org/10.1016/j.media.2022.102471.

[4] S. M. Smith, "Overview of fMRI analysis," The British Journal of Radiology, vol. 77, no. suppl_2, pp. S167–S175, Dec. 2004, doi: https://doi.org/10.1259/bjr/33553595.

[5] K. Zheng, S. Yu, L. Chen, L. Dang, and B. Chen, "BPI-GNN: Interpretable brain network-based psychiatric diagnosis and subtyping," NeuroImage, vol. 292, pp. 120594–120594, Apr. 2024, doi: https://doi.org/10.1016/j.neuroimage.2024.120594.

[6] Richardson, H., Lisandrelli, G., Riobueno-Naylor, A. et al. Development of the social brain from age three to twelve years. Nat Commun 9, 1027 (2018).

[7] Lebo Wang, Kaiming Li, Xiaoping P. Hu; Graph convolutional network for fMRI analysis based on connectivity neighborhood. Network Neuroscience 2021; 5 (1): 83–95. doi: https://doi.org/10.1162/netn$_{a0}$0171