DGL-24 Kaggle Competition: Your Link Logic

Gita Salsabila *02518229*
Amani Aldahmash *01836401*
Kangle Yuan *02472956*
Mengyu Rao *02443661*
Jiqiu Hu *02429885*

Introduction

The field of brain science constantly seeks new ways to understand how different brain regions connect and communicate. However, brain image quality can vary due to environmental conditions or imaging technologies. High-quality images provide detailed information but are challenging to obtain, while low-quality images lack detail but are more affordable. This issue has led to the development of methods like Graph Super-Resolution (GSR-Net) [2] to enhance brain image quality. GSR-Net focuses on brain connectivity represented as graphs, where nodes represent brain regions and edges represent connections. By leveraging this connectivity, GSR-Net can capture complex relationships in graph data and improve low-resolution brain connectivity maps. In our research, we aimed to build an improved version of GSR to predict high-resolution connectivity graphs from low-resolution brain graphs in an inductive learning setting.

Datasets

Our work utilizes the pre-processed dataset derived from the SLIM functional MRI dataset [1]. This pre-processed dataset comprises two sets of one-dimensional arrays for each resolution type, designated as the Low-Resolution (LR) dataset and the High-Resolution (HR) dataset. Each one-dimensional data represents a set of encoding for brain connections. These arrays are generated from the original dataset through the vectorisation process method. Each dataset contains 279 samples, which are subsequently divided into a Training set (167 examples), a Public test set (56 examples) and a Private test set (56 examples). Only data in the Training set is available for us to train and validate our model, and the performance of our model will be tested using the Public test set and the Private test set during the competition. Given that our model will be trained using the entire brain graph matrix, we employ the anti-vectorisation method for further pre-processing. We then split the pre-processing Training set into the internal training set (133 samples) and the internal testing set (34 samples).

Methods

We proposed an advanced brain network framework called the **Attention-based Graph Super-Resolution (Att-GSR)** as shown in Figure 3 in Appendix A. In the development of the Att-GSR model, we made strategic modifications to the baseline GSR-Net model [2] to enhance its performance on our task. We have adapted 2 principal components, each adapted to improve the model's efficiency.

A. *Graph UNet with GAT*

Replacing the standard GCN with a GAT was one of the pivotal changes that allowed the model to capture relational dependencies using an attention mechanism. This change allowed the model to focus on interactions between nodes and made the model understand the graph structure better. It also can allocate different levels of importance and attention to each node for a better feature representation of the connectivity patterns. This inclusion of multiple GAT layers allowed for the incorporation of pooling and unpooling, which enables the model to capture hierarchical features at different resolutions which is important for understanding the multi-scale brain structure. It also enabled the model to align the output graph structure with the targeted size of 268x268. We set the output graph of the model to 268x268 rather than 320x320 as in the GSR-Net [2] model. This step not only simplified the pipeline by eliminating the need for padding but also reduced the sparsity of the feature maps that were learnt with the model, which can lead to more accurate results. The self-reconstruction loss (L1) in this layer ensures that the resulting features reflect the LR data's structure.

B. *Modified GSR Layer*

We have refined this layer to enhance the learning process by switching to the AdamW optimiser, which introduced decoupled weight decay for regularization and led to better generalization capabilities in our Att-GSR model. Inspired by the DropGNN paper [3], we also integrated dropout layers into the GAT and convolution layers to boost the graph expressiveness and to inject noise that protects against overfitting. This layer utilizes trainable filters and eigen-decomposition losses(L3) to ensure that the super-resolved results adhere to the characteristics of the HR graph.

C. *Training and Testing*

Our model shows more efficiency through a reduced number of trainable parameters up to 24.37%, making Att-GSR (504.108 params) more lightweight than the GSR-Net baseline model (666.560 params). Another important modification that we used was adding a batching mechanism during training, where we set our batch size to 2, which helped with regularization. We set the number of epochs to 200, learning rate 0.0001, and we use Xavier Glorot following the number in the original GSR paper [2]. This provides the model with more opportunities to learn from the data and achieve a better performance.

During training, our model uses the MSE loss which combines the network's immediate output, the initial features, and the eigenvalues of the model's weights and target graph, as detailed in Appendix B, which is similar to the loss function of GSR-Net model [2]. We adjusted the GSR loss function by introducing a new scaling factor to the Generation Loss term. This change increases the penalty for the GAT module when it fails to predict HR graphs from the given LR graph. This composite loss ensures that Att-GSR learns the accurate representation of the brain graph as well as adhering to any underlying spectral properties. While on testing, Att-GSR evaluates its performance on unseen low-resolution brain graphs, generating high-resolution ones and comparing them to the ground truth. This process involves measuring the MSE to measure the accuracy of these new results.

D. Permutation invariance and equivariance properties

In the Att-GSR model, we carefully consider the fundamental principles of permutation invariance and equivariance. This is achieved through the use of GAT layers, which are designed to not rely on the order of nodes in the input graph but instead focus on their connections. The attention mechanism in these layers assigns weights to node connections based on features rather than positions. Consequently, even if the input nodes are rearranged, the learned representation, features, and connections remain consistent.

Results and Discussion

We evaluate our model using 6 different metrics and use 3-fold cross-validation. For different folds, we calculated the values of 6 different metrics. Figure 1 show the average of the results of 6 different metrics at 3-fold, and we use error lines to represent the standard deviation across the folds. For more detail, please refer to Figure 4 in Appendix C.

We can conclude that among all evaluation metrics, the model performs best on the MAE (PC) and the MAE (EC) metrics. The MAE (BC) is slightly higher, around 0.11. These values mean that the model performs well in predicting the importance of network nodes. Moreover, the error lines on these three bars are small, which means the model shows high stability and accuracy in predicting Centrality metrics (BC, EC and PC).

The model's performance on MAE, PCC, and JSD is not as good as previously analyzed metrics. When using 3-fold cross-validation, these metrics show fluctuations. For MAE, values range between 0.26 to 0.31 across folds, indicating unstable prediction accuracy. PCC, which is measuring linear correlation, averages around 0.21, suggesting a moderate linear relationship in predictions. JSD scores, which measure distribution similarity, average around 0.13, with short error lines indicating similarity between predicted and actual distributions. Overall, the model is good in predicting network centrality metrics but fluctuates in predicting connection strength (MAE) and linear relationship (PCC), showing a stronger and more stable understanding of centrality features.

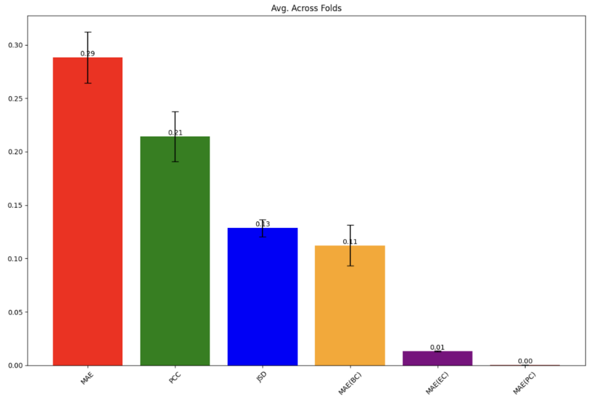

Fig. 1. The average of the results of 6 different metrics at 3-fold

The table I displays the training time and RAM usage, which remain relatively consistent during 3-fold cross-validation when the model is trained using a device with 30 GiB RAM, 8-core CPU, and 16 GiB GPU NVIDIA Quadro. From this, we can conclude that Att-GSR is an efficient model, as it trains relatively quickly, taking only 12 on average, and utilizes only 2.7 GB of memory during each training session.

TABLE I
THE TRAINING TIME FOR 3F-CF AND AVERAGE RAM USAGE.

3F-CF	training time	RAM usage
Fold 1	733.34 seconds	2.6773 GB
Fold 2	732.92 seconds	2.7592 GB
Fold 3	742.71 seconds	2.7935 GB
Average	736.32 seconds	2.7433 GB
Total	2208.97 seconds	8.2299 GB

We re-train the model on the whole pre-processed Training set containing 167 samples. To better demonstrate the performance of our model, we randomly choose an LR matrix and use Heatmap to display the ground truth of the HR matrix and our predicted HR matrix, shown in Figure -D. We find that the two heatmaps are quite similar, indicating that the Att-GSR model is most likely to perform well.

Our model secured 9th position in the Kaggle competition: Brain Graph Super-Resolution Challenge 2024 based on private test score (MAE = 0.149390).

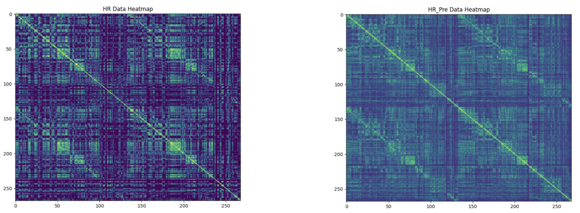

Fig. 2. Ground Truth vs Predicted HR Graph Heatmap

References

[1] W. Liu, D. Wei, Q. Chen, W. Yang, J. Meng, G. Wu, T. Bi, Q. Zhang, X.-N. Zuo, and J. Qiu. Longitudinal test-retest neuroimaging data from healthy young adults in southwest china. Scientific data, 4(1):1–9, 2017.

[2] M. Isallari, I. Rekik. GSR-Net: Graph Super-Resolution Network for Predicting High-Resolution from Low-Resolution Functional Brain Connectomes, 2010.

[3] P. András Papp, K. Martinkus, L. Faber, R. Wattenhofer. DropGNN: Random Dropouts Increase the Expressiveness of Graph Neural Networks, 2021.

[4] Veličković, Petar, et al. "Graph attention networks."arXiv preprint arXiv:1710.10903(2017).

Appendix A
Model Architecture

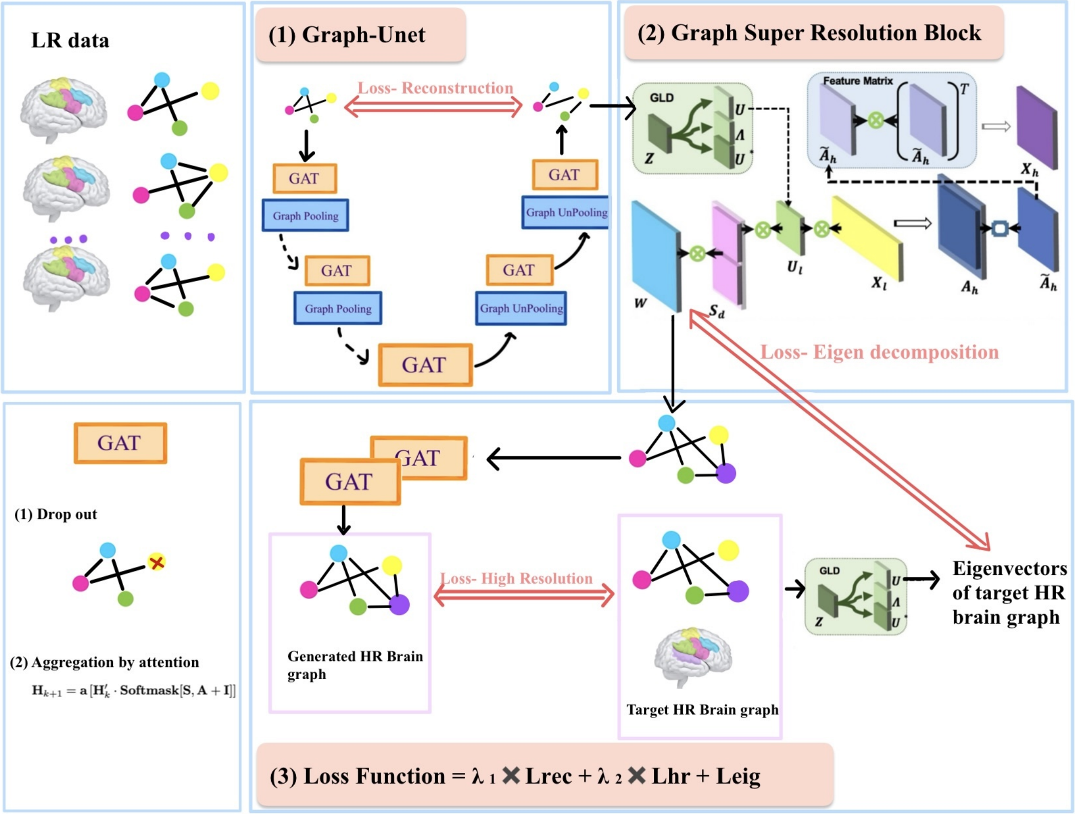

Fig. 3. Att-GSR Model Pipeline

Appendix B
Loss function $\mathcal{L}$

$$
\begin{aligned}
\mathcal{L} &= \lambda_1 \mathcal{L}_{\text{rec}} + \lambda_2 \mathcal{L}_{\text{hr}} + \mathcal{L}_{\text{eig}} \\
&= \frac{1}{N}\sum_{i=1}^{N} ||Z_{0i} - Z_{1i}||_2^2 + \lambda_2 \frac{1}{N}\sum_{i=1}^{N} ||Z_{hi} - A_{hi}||_2^2 + \frac{1}{N}\sum_{i=1}^{N} ||W_i - U_{1i}||_2^2
\end{aligned}
$$

$\mathcal{L}_{\text{rec}}$: **Reconstruction Loss**
Z_{0i} : initial node feature embedding of the LR connectome
Z_{1i} : the output of the Graph UNet

$\mathcal{L}_{\text{hr}}$: **Generation Loss**
Z_{hi} : predicted HR graph (result of GAT layer)
A_{hi} : target HR graph

$\mathcal{L}_{\text{eig}}$: **Eigendecomposition Loss**
W_i : learnable parameters in this GSR layer
U_{1i} : Eigenvectors of the ground truth HR graph

APPENDIX C
3 FOLDS CROSS VALIDATION PLOTS

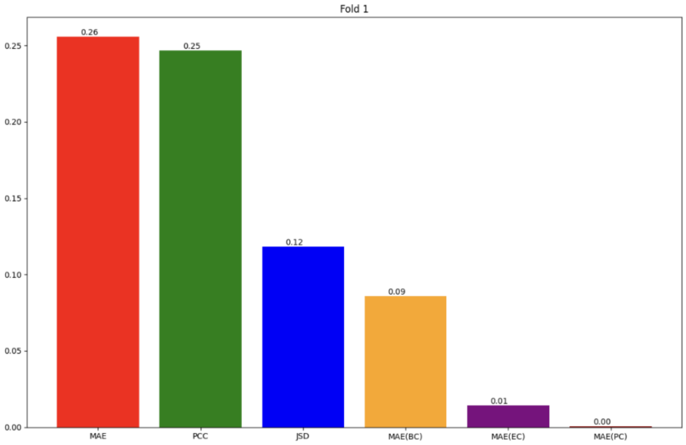

Fig. 4. The results of 6 different metrics at Fold 1

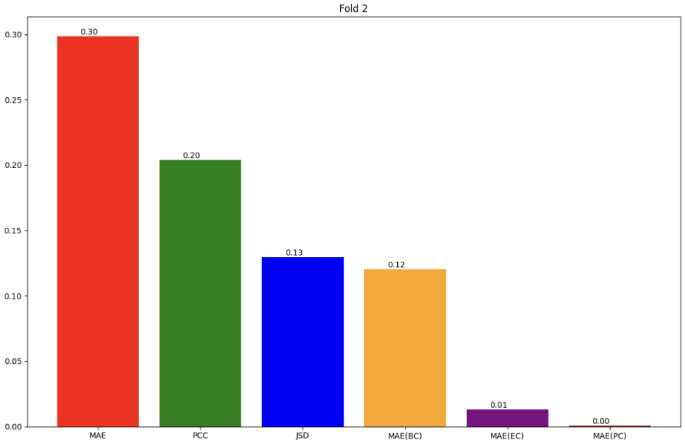

Fig. 5. The results of 6 different metrics at Fold 2

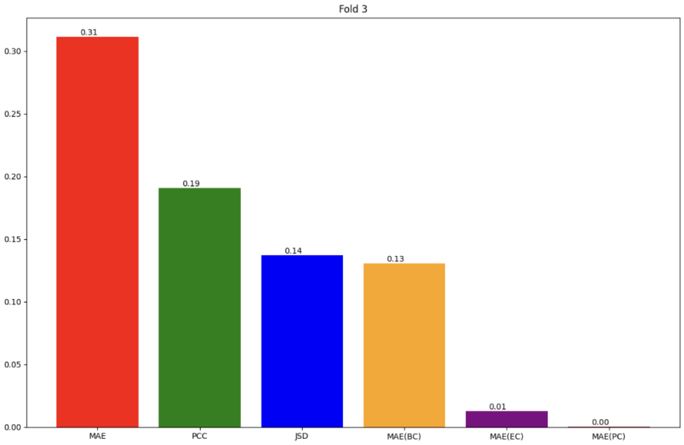

Fig. 6. The results of 6 different metrics at Fold 3