How Much to Aggregate: Learning Adaptive Node-Wise Scales on Graphs for Brain Networks

Injun Choi[1], Guorong Wu[2], and Won Hwa Kim[1,3(✉)]

[1] Pohang University of Science and Technology, Pohang, South Korea
{surung9898,wonhwa}@postech.ac.kr
[2] University of North Carolina at Chapel Hill, Chapel Hill, USA
grwu@med.unc.edu
[3] University of Texas at Arlington, Arlington, USA

Abstract. Brain connectomes are heavily studied to characterize early symptoms of various neurodegenerative diseases such as Alzheimer's Disease (AD). As the connectomes over different brain regions are naturally represented as a graph, variants of Graph Neural Networks (GNNs) have been developed to identify topological patterns for disease early diagnosis. However, existing GNNs heavily rely on the fixed local structure given by an initial graph as they aggregate information from a direct neighborhood of each node. Such an approach overlooks useful information from further nodes, and multiple layers for node aggregations have to be stacked across the entire graph which leads to an over-smoothing issue. In this regard, we propose a flexible model that learns adaptive scales of neighborhood for individual nodes of a graph to incorporate broader information from appropriate range. Leveraging an adaptive diffusion kernel, the proposed model identifies desirable scales for each node for feature aggregation, which leads to better prediction of diagnostic labels of brain networks. Empirical results show that our method outperforms well-structured baselines on Alzheimer's Disease Neuroimaging Initiative (ADNI) study for classifying various stages towards AD based on the brain connectome and relevant node-wise features from neuroimages.

1 Introduction

Rich bodies of works show that amyloid deposition and neurofibrillary tangles damage neural connections in the brain, suggesting the analysis of brain connectomes in neuroimaging studies to characterize early symptoms of brain disorders such as Autism [1], Parkinson's Disease (PD) [24] and Alzheimer's Disease (AD) [6,20]. The connectome connects different anatomical regions of interest (ROI) in the brain and comprises a brain network for individual subjects. Such a brain

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-16431-6_36.

L. Wang et al. (Eds.): MICCAI 2022, LNCS 13431, pp. 376–385, 2022.
https://doi.org/10.1007/978-3-031-16431-6_36

network is mathematically represented as a graph that is defined by a set of nodes and edges, whose nodes are given by the ROIs and the connectomes define edges as a measure of strength among the nodes derived from structural and functional neuroimages, e.g., tractography on Diffusion Tensor Images (DTI).

Such a graph representation of a brain network, together with image-derived measurements at each ROI, naturally justifies the utilization of graph deep learning approaches such as graph neural network (GNN) for disease characterization. GNN [9] and its variants [4,13] incorporate the structure of graphs via message passing among connected nodes in the graph. To obtain more robust features, they aggregate direct neighborhood information and refer to indirectly connected nodes by stacking multiple aggregation layers, which lead to promising results in node classification [7], link prediction [26] under the homophily condition [15], i.e., adjacent nodes have high similarity as in adjacent pixels in natural images.

However, there are still several issues for previous GNNs to be adopted for brain network analysis. First, they often use a single graph merely as the domain of measurements at each node. Also, the graph is often represented as a binary matrix, which does not incorporate exact relationships in the neighborhood of its node. Of course there have been recent efforts to alleviate these problems such as Graph Attention Network (GAT) [22], Graph Convolutional using Heat Kernel (GraphHeat) [25] and Graph Diffusion Convolution (GDC) [14]. However, they cannot incorporate heterogeneous characteristics of brains, where both ROI measures and brain networks are different across subjects. A bigger problem is that these methods are either too local or global: aggregation of information occurs only within the direct neighbors of each node and adding layers to incorporate indirect neighbors triggers the local aggregation across all the nodes.

The issues above naturally lead to an idea of learning adaptive range for individual nodes. As each brain ROI has different biological and topological properties, it is feasible to learn different local receptive fields that provide an understanding of subnetwork structure. For this, we propose a novel flexible framework that learns suitable scales of each node's neighborhood by leveraging a diffusion kernel and a specialized model architecture to update the scale as a parameter in the model. Learning individual scales for each node lets our model find the right spatial range for information propagation for each ROI.

Key Contributions: Our work leads to 1) learning adaptive local neighborhood to aggregate information for better prediction of graph labels, 2) deriving a parametric formulation to perform gradient-based learning on local receptive field of nodes using a diffusion kernel, and 3) validating the developed framework in comparisons to the recent graph neural network models. Experiments on structural brain networks from Diffusion Tensor Imaging (DTI) and ROI measures from functional imaging from Alzheimer's Disease Neuroimaging Initiative (ADNI) study show that the developed framework demonstrates superior graph-level classification results identifying the independence of each ROI.

2 Related Work

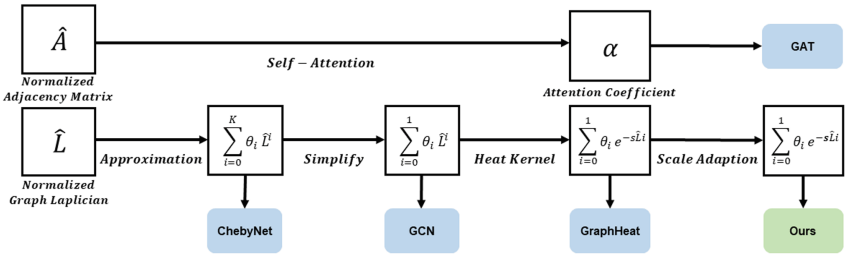

Fig. 1. An overview of node aggregation frameworks. Different from the previous methods, our model can flexibly learn the scale s for each node. Note that s in the GraphHeat is a constant hyperparameter.

To generalize the Convolutional Neural Networks (CNNs) to signals defined on graphs, various spectral methods such as Graph Convolutional Network and ChebyNet were proposed in [2,4,11,13], allowing the use of shared filters. In these models, the importance of each node is given dichotomously, limiting the selection of proper nodes in the neighborhood. To address this issue, the Graph Convolutional Networks using Heat Kernel (GraphHeat) [25] uses heat diffusion to quantify relationships among nodes, Graph Attention Network (GAT) [22] generates the importance score of different nodes by using attention mechanism, and Graph Diffusion Convolution (GDC) [14] uses a transition matrix utilizing personalized PageRank node proximity measure. A streamline of these methods is introduced in Fig. 1, and are discussed in detail in the following section.

3 Preliminaries

Graph Convolution. An undirected graph $G = \{V, E\}$, where V is a node set with $|V| = N$ and E is a set of edges, has an symmetric adjacency matrix $A_{N\times N}$ encoding node connectivity. Graph Laplacian $L = D - A$, where D is a diagonal degree matrix of G with $D_{i,i} = \sum_j A_{i,j}$, and a normalized graph Laplacian is defined as $\hat{L} = I_N - D^{-1/2}AD^{-1/2}$, where I_N is an identity matrix. Since $\hat{L}$ is real and symmetric, $\hat{L}$ has a complete set of orthonormal basis $U = [u_1|u_2|...|u_N]$ and corresponding real and non-negative eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_N$.

With U, the Graph Fourier transform of signal x is defined as $\hat{x} = U^{\mathcal{T}}x$ and its inverse transform is $x = U\hat{x}$, where $\hat{x}$ is the signal in Graph Fourier space [19]. By the convolution theorem [16], a graph convolution operation is defined as:

$$g * x = U((U^{\mathcal{T}}g) \circ (U^{\mathcal{T}}x)), \tag{1}$$

where g is a filter and $\circ$ is hadamard product.

Graph Convolutional Network. In a spectral graph convolutional network [2], spectral convolution between a filter g and a signal x on a graph was defined as:

$$g * x = U g_\theta U^\mathcal{T} x = (\theta_1 u_1 u_1^\mathcal{T} + \theta_2 u_2 u_2^\mathcal{T} + ... + \theta_N u_N u_N^\mathcal{T}) x \tag{2}$$

where $U^\mathcal{T} g$ in Eq. (1) is replaced by a kernel $g_\theta = diag(\{\theta_i\}_{i=1}^N)$. As using Eq. (2) can be computationally expensive, polynomial approximation using Chebyshev expansions was proposed [4]. ChebyNet provides a polynomial filter $g = \sum_{k=0}^{K-1} \theta_k \Lambda^k$ where the parameter $\theta \in \mathbb{R}^K$ with $K \ll N$ is vector of polynomial coefficients. With a polynomial filter g, graph convolution is performed as:

$$g * x \simeq (\theta_0 I + \theta_1 \hat{L} + ... + \theta_{K-1} \hat{L}^{K-1}) x. \tag{3}$$

Later, GCN [13] was proposed with only the first-order approximation as:

$$g * x \simeq \theta (I - \hat{L}) x, \tag{4}$$

which is a simplified version of ChebyNet.

Heat Kernel on Graphs. In [3], heat kernel between nodes p and q of a graph G is defined in the spectral graph domain as:

$$h_s(p, q) = \sum_{i=1}^{N} e^{-s\lambda_i} u_i(p) u_i(q) \tag{5}$$

where λ_i and u_i are the i-th eigenvalue and eigenvector of the graph Laplacian, and s controls the time/scale of diffusion. Later in [25], the GraphHeat generates the connectivity measure using heat-kernel, and the similarity via the heat diffusion replaces binary adjacency matrix for GNN to capture more precise relationships. Since GraphHeat only retains the first two terms in Eq. (3) for efficiency, the convolution with heat kernel is approximated as:

$$h_s * x \simeq (\theta_0 I + \theta_1 e^{-s\hat{L}}), \tag{6}$$

where h_s acts as a low-pass filter.

4 Method: Learning Node-Wise Adaptive Scales

In this section, we propose a flexible model to learn the range of adaptive neighborhoods for each graph node to capture the optimal local context to improve a downstream prediction task. Figure 2 explains the fundamental idea where nodes n_1 and n_2 aggregate information from their neighborhoods. The left panel shows an example where n_2 (blue) is misclassified as red when the equal receptive field of s_1 is used both n_1 and n_2, whereas n_1 is misclassfied as blue if s_2 is used. Therefore, s_1 and s_2 need to be applied adaptively for n_1 and n_2 so that the right local neighborhoods are selected for individual nodes for data aggregation. Manual selection of these scales can be extremely exhaustive, thus it requires a specialized model to "learn" the scales in a data-driven way.

In the following, we consider a graph classification problem. The objective is to learn to predict a graph-wise label y for input $G = \{A, X\}$, where A is an

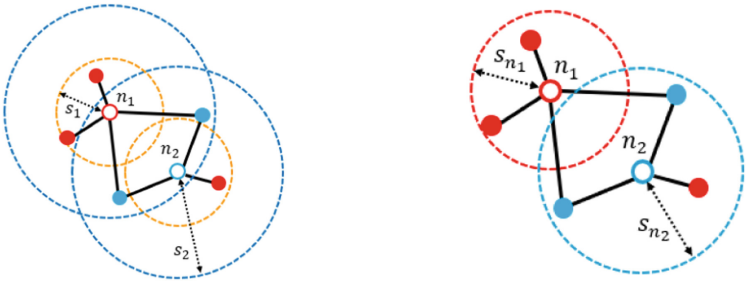

Fig. 2. An example of adaptive neighborhood for nodes n_1 and n_2. Node color (red and blue) denotes node class. Left: Applying the same scale (s_1 or s_2) leads to false aggregation. Right: adaptively applying s_{n_1} and s_{n_2} leads to a proper aggregation. (Color figure online)

adjacency matrix and X is node-wise feature. We first introduce a model that learns a single s globally for a graph, then design a model that learns individual s_i for each node of the graph expecting that adaptively aggregating X through the graph structure will significantly improve the prediction performance.

Adaptive Convolution via Scale. While [25] defines the scale s as a hyper-parameter for an entire graph, we propose a model that trains on the s. The objective function is defined by cross-entropy between the true value Y_{tc} at the c-th class for the t-th sample and the model prediction $\hat{Y}_{tc}$, and a regularizer on s to prevent it from becoming negative:

$$\mathcal{L}(s) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{c\in C} Y_{tc}\log \hat{Y}_{tc}(s) + \lambda|s|[s<0], \tag{7}$$

where λ is a hyperparameter, T is a sample size, and $|s|[s<0]$ takes an absolute value of s when the scale becomes less than 0. Update of the scale is performed as $s \leftarrow s - \alpha_s \frac{\partial \mathcal{L}}{\partial s}$ with a learning rate α_s for s via gradient-based methods along with other learnable parameters W. Derivation of $\frac{\partial \mathcal{L}}{\partial s}$ is shown below.

Forward Propagation. Our model consists of multiple graph convolution layers that adaptively aggregate information for each node and an output layer that predicts a class label for an input graph. From Eq. (6), each of graph convolution layer is defined with a non-linear activation function σ_k as:

$$H_k = \sigma_k(e^{-s\hat{L}}H_{k-1}W_k), \tag{8}$$

where H_k is an output from the k-th convolution layer with $H_0 = X$, and W_k is a matrix of learnable parameters. To obtain a prediction $\hat{Y}_{tc}$, given K graph convolution layers, the output H_K is vectorized and applied with a readout function $\psi(\cdot)$ (e.g., Multi-Layer Perceptron) to obtain integrated values for predicting each class in C. Finally, $Softmax$ is used to get class-wise pseudo-probability as

$$\hat{Y}_{tc} = \frac{\psi(H_K)_{tc}}{\sum_{c'\in C}\psi(H_K)_{tc'}}, \tag{9}$$

which is fed into Eq. (7) for training.

Training on the Scale. From Eq. (5) and Eq. (7)-(9), we can calculate the derivative of $\mathcal{L}$ in terms of scale parameter s. For simplicity, let us consider a single sample case with $T = 1$ and an arbitrary kernel $\mathcal{K}(s)$. In Eq. (7), let the error term be $\mathcal{L}_{ce}$ and the regularization be $\mathcal{L}_{scale}$. First, the derivative of $\mathcal{L}_{scale}$ with respect to s is:

$$\frac{\partial \mathcal{L}_{scale}}{\partial s} = -\lambda[s < 0]. \tag{10}$$

Now, the derivative of $\mathcal{L}_{ce}$ w.r.t. s can be derived using chain rule as:

$$\frac{\partial \mathcal{L}_{ce}}{\partial s} = \frac{\partial \mathcal{L}_{ce}}{\partial \psi(H_K)} \frac{\partial \psi(H_K)}{\partial H_K} \frac{\partial H_K}{\partial s}, \tag{11}$$

where

$$\frac{\partial \mathcal{L}_{ce}}{\partial \psi(H_K)} = \hat{Y} - Y, \frac{\partial H_k}{\partial s} = \sigma_k'(\mathcal{K}(s)H_{k-1}W_k)^\mathcal{T} W_k^\mathcal{T} (H_{k-1}^\mathcal{T} \mathcal{K}'(s) + \frac{\partial H_{k-1}}{\partial s}\mathcal{K}(s)).$$

As $\psi'(H_K)$ depends on the choice of $\psi(H_K)$, the final gradient on the loss is:

$$\frac{\partial \mathcal{L}}{\partial s} = \sum_{i=1}^{N}\sum_{j=1}^{N}(\frac{\partial \mathcal{L}_{ce}}{\partial H_k}\sigma_k'(\mathcal{K}(s)H_{k-1}W_k)^\mathcal{T} W_k^\mathcal{T} (H_{k-1}^\mathcal{T} \mathcal{K}'(s) + \frac{\partial H_{k-1}}{\partial s}\mathcal{K}(s)))_{ij} - \lambda[s < 0] \tag{12}$$

where $i, j \in \{1, ..., N\}$ denoting the i-th and the j-th node, and $\frac{\partial \psi(H_K)}{\partial H_K}$ is embedded in $\frac{\partial \mathcal{L}_{ce}}{\partial H_K}$ for all $k \in \{1, ..., K\}$. Note that s is univariate and covers the entire graph. The full derivation of Eq. (12) is shown in the supplementary.

Localization to Each Node. Figure 2 (right) shows that node n_1 and n_2 having adaptive neighborhood size s_{n_1} and s_{n_2} can capture more precise information. Therefore, we propose a diffusion model to train on the local receptive fields (i.e., scale) for each node. We directly update each scale by removing the marginalization over the nodes (i.e., $\sum_{i=1}^{N}$) in the cross-entropy term of Eq. (12) as:

$$\frac{\partial \mathcal{L}_{ce}}{\partial s_i} = \frac{1}{T}\sum_{t=1}^{T}\sum_{j=1}^{N}(\frac{\partial \mathcal{L}_{ce}}{\partial H_k}\sigma_k'(\mathcal{K}(s_i)H_{k-1}W_k)^\mathcal{T} W_k^\mathcal{T} (H_{k-1}^\mathcal{T} \mathcal{K}'(s_i) + \frac{\partial H_{k-1}}{\partial s_i}\mathcal{K}(s_i)))_{tij} \tag{13}$$

where $i \in \{1, ..., N\}$. The s_i is given for each node n_i and can be trained with gradient-based methods.

5 Experiment

Dataset. Total of 401 subjects with diffusion-weighted imaging (DWI), Amyloid-PET and FDG-PET were taken from pre-selected ADNI cohort. Each brain was partitioned into 148 cortical surface regions using Destrieux atlas [5],

Table 1. Demographics of the ADNI dataset.

Category	CN	SMC	EMCI	LMCI	AD
# of Subjects	89	53	132	55	72
Gender (M/F)	37/52	19/34	84/48	32/23	42/30
Age (Mean ± std)	72.6 ± 4.8	73.4 ± 4.8	70.3 ± 7.1	72.3 ± 6.2	75.8 ± 7.2

and tractography on DWI was applied to calculate the white matter fibers connecting the brain regions to construct 148×148 structural network. On the same parcellation, region-wise imaging features such as SUVR (standard uptake value ratio) of β-amyloid protein from Amyloid-PET, SUVR of metabolism level from FDG-PET, cortical thickness from MRI, and nodal degree were defined. Cerebellum was used as the reference for SUVR normalization. The diagnostic labels for each subject were defined as cognitive normal (CN), significant memory concern (SMC), early mild cognitive impairment (EMCI), late mild cognitive impairment (LMCI) and AD. The demographics of the ADNI dataset is given in Table 1.

Table 2. Performance Comparisons of Various GNN Models on ADNI Data.

Feature	Methods	Accuracy	Precision	Specificity	Sensitivity (Recall)
Degree	SVM	0.532 ± 0.162	0.509 ± 0.109	0.886 ± 0.037	0.616 ± 0.220
	GCN	0.466 ± 0.079	0.422 ± 0.085	0.867 ± 0.019	0.461 ± 0.107
	GAT	0.633 ± 0.093	0.610 ± 0.085	0.908 ± 0.025	0.681 ± 0.101
	GraphHeat	0.641 ± 0.077	0.624 ± 0.071	0.908 ± 0.021	0.672 ± 0.083
	GDC	0.670 ± 0.090	0.660 ± 0.088	0.917 ± 0.024	0.684 ± 0.103
	Ours (global t)	**0.703 ± 0.068**	**0.671 ± 0.079**	**0.925 ± 0.017**	**0.744 ± 0.072**
	Ours (local t)	0.653 ± 0.076	0.620 ± 0.068	0.913 ± 0.022	0.690 ± 0.101
Cortical Thickness	SVM	0.721 ± 0.051	0.669 ± 0.060	0.933 ± 0.016	0.862 ± 0.050
	GCN	0.494 ± 0.076	0.464 ± 0.073	0.867 ± 0.024	0.518 ± 0.098
	GAT	0.865 ± 0.111	0.865 ± 0.098	0.966 ± 0.028	0.874 ± 0.110
	GraphHeat	0.828 ± 0.056	0.843 ± 0.050	0.956 ± 0.014	0.853 ± 0.055
	GDC	0.860 ± 0.063	0.871 ± 0.061	0.965 ± 0.016	0.878 ± 0.055
	Ours (global t)	0.841 ± 0.106	0.848 ± 0.112	0.960 ± 0.024	0.865 ± 0.090
	Ours (local t)	**0.875± 0.043**	**0.873 ± 0.046**	**0.968 ± 0.011**	**0.896 ± 0.032**
β-Amyloid	SVM	0.843 ± 0.093	0.819 ± 0.089	0.961 ± 0.025	0.882 ± 0.084
	GCN	0.526 ± 0.069	0.499 ± 0.074	0.880 ± 0.019	0.535 ± 0.106
	GAT	0.873 ± 0.057	0.876 ± 0.055	0.968 ± 0.015	0.889 ± 0.053
	GraphHeat	0.881 ± 0.081	0.878 ± 0.095	0.970 ± 0.020	0.877 ± 0.114
	GDC	0.893 ± 0.108	0.875 ± 0.151	0.974 ± 0.025	**0.915 ± 0.077**
	Ours (global t)	0.911 ± 0.072	0.912 ± 0.085	0.977 ± 0.017	0.911 ± 0.085
	Ours (local t)	**0.916 ± 0.078**	**0.912 ± 0.093**	**0.979 ± 0.019**	0.914 ± 0.099
FDG	SVM	0.853 ± 0.044	0.829 ± 0.053	0.964 ± 0.011	0.919 ± 0.029
	GCN	0.511 ± 0.066	0.474 ± 0.088	0.876 ± 0.017	0.535 ± 0.108
	GAT	0.678 ± 0.089	0.673 ± 0.102	0.919 ± 0.024	0.685 ± 0.105
	GraphHeat	0.885 ± 0.065	0.893 ± 0.067	0.971 ± 0.016	0.902 ± 0.065
	GDC	0.923 ± 0.089	0.923 ± 0.104	0.980 ± 0.024	0.949 ± 0.052
	Ours (global t)	0.928 ± 0.067	0.931 ± 0.078	0.982 ± 0.017	0.945 ± 0.050
	Ours (local t)	**0.960 ± 0.028**	**0.963 ± 0.031**	**0.990 ± 0.007**	**0.965 ± 0.028**
All Imaging Features	SVM	0.935 ± 0.042	0.917 ± 0.048	0.985 ± 0.010	0.953 ± 0.037
	GCN	0.556 ± 0.074	0.537 ± 0.065	0.888 ± 0.018	0.562 ± 0.126
	GAT	0.726 ± 0.073	0.710 ± 0.070	0.932 ± 0.019	0.746 ± 0.077
	GraphHeat	0.923 ± 0.047	0.923 ± 0.050	0.980 ± 0.012	0.931 ± 0.043
	GDC	0.930 ± 0.066	0.930 ± 0.073	0.983 ± 0.016	0.945 ± 0.052
	Ours (global t)	0.933 ± 0.056	0.933 ± 0.057	0.983 ± 0.015	0.945 ± 0.044
	Ours (local t)	**0.953 ± 0.032**	**0.955 ± 0.035**	**0.988 ± 0.008**	**0.957 ± 0.029**

Setup. We trained a two-layer graph convolution model with 16 hidden units, and used a rectified linear unit (ReLU) for the activation function. ψ was a two-level linear layers, and dropout with predefined rate for each model. Weights were initialized with Xavier initialization [8] and trained with Adam optimizer [12] at the learning rate of 0.01. The α_s was chosen as either 0.1 or 0.01. We used the kernel $\mathcal{K}(\hat{L}, s)$ as Heat kernel defined in Eq. (5). Heat kernel's initial scale was $s = 2$ for both global and local models, and heat kernel values were thresholded at $< 1e-5$. Regularization λ was 1. 10-fold cross-validation (CV) was used and accuracy/precision/specificity/sensitivity in average were computed for evaluation. One-vs-rest scheme was used to compute the evaluation metrics.

Support Vector Machine (SVM) as well as recent GNN models such as GCN [13], GAT [22], GraphHeat [25], and GDC [14] were adopted as the baseline models. Each baseline was set up and trained at our best effort to obtain feasible outcomes for fair comparisons. More details are given in the supplementary.

Quantitative Results. All results are reported in Table 2 at a glance. It shows that our models (training both global and local s) empirically outperform in all experiments except the recall of classification with β-amyloid. The highest accuracy of 96% in classifying the 5 classes was achieved with the metabolism (FDG) on the structural network, which is known to be an effective biomarker for characterizing early AD. The standard deviation on all evaluation measures stayed low across the 10-folds addressing our models' stability. Especially comparing the results from GraphHeat and our models proves that training on the scale definitely improves the results where the scale (hyper-)parameter for the GraphHeat was used as the initialization for our models. As the sample-size was small, SVM worked efficiently, and the GNN models other than GCN showed good performance; this may be because these GNN models extract weighted adjacency matrix but binary adjacency matrix (thresholded from the brain network) was used for the GCN. Adopting all features was mostly better but underperformed FDG measure with training with local scale. This may be because cortical thickness is not a suitable biomarker to discriminate very early stages of AD.

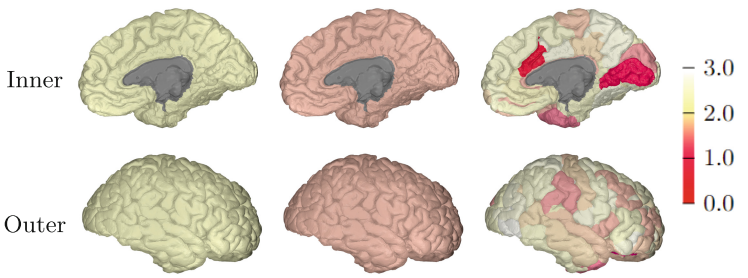

Fig. 3. A visualization of learned scales on the right hemisphere of a brain and localized ROIs. Left: initial scale ($s = 2$), Middle: globally trained scale ($s = 1.59$), Right: locally trained scales, Bottom: Region of interests having lower scale than 1 in the local model.

Qualitative Results. In Fig. 3, we visualize the initial scale ($s = 2$ for GraphHeat), globally trained scale ($s = 1.586$), and adaptively trained scale for each ROI. The ROIs with small scales denote that it is independent: they do not require node aggregation from large neighborhoods to improve the AD classification. Table in Fig. 3 shows those ROIs with the trained $s < 1$. Interestingly, middle-anterior cingulate gyrus and sulcus from both left and right hemispheres showed the smallest scales (0.171 and 0.207) demonstrating the highest locality. It is responsible for cognitive and executive functions reported in [10,21] to be AD-specific. Other ROIs with low scales such as temporal lingual gyrus, inferior temporal and post central regions are also consistently found in preclinical AD [17,18,23], which demonstrate that these were the key ROIs in discriminating even early stages of Alzheimer's disease in our model.

6 Conclusion

In this paper, we proposed a novel model that flexibly learns individual node-wise scales in a brain network to adaptively aggregate information from neighborhoods. The developed model lets one identify which ROIs in the brain behave locally (i.e., independently) on the brain network structure to predict global diagnostic labels. We have derived a rigorous formulation of the scale such that it can be trained via gradient-based method, and validated that our model can accurately classify AD-specific labels of brain networks and detect key ROIs corroborated by other AD literature.

Acknowledgements. This research was supported by NSF IIS CRII 1948510, NIH R03 AG070701 and partially supported by IITP-2019-0-01906 (AI Graduate Program at POSTECH), IITP-2022-2020-0-01461 (ITRC) and IITP-2022-0-00290 funded by Ministry of Science and ICT (MSIT).

References

1. Anderson, J., Nielsen, J., Froehlich, A., et al.: Functional connectivity magnetic resonance imaging classification of autism. Brain **134**(12), 3742–3754 (2011)
2. Bruna, J., Zaremba, W., Szlam, A., Lecun, Y.: Spectral networks and deep locally connected networks on graphs. In: ICLR, Banff, Canada (2014)
3. Chung, F.: Spectral graph theory. American Mathematical Society (1997)
4. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: NIPS, vol. 29, pp. 3844–3852 (2016)
5. Destrieux, C., Fischl, B., Dale, A., Halgren, E.: Automatic parcellation of human cortical gyri and sulci using standard anatomical nomenclature. Neuroimage **53**(1), 1–15 (2010)
6. DeTure, M., Dickson, D.: The neuropathological diagnosis of Alzheimer's disease. Mol. Neurodegeneration **14**(32), 1–18 (2019)
7. Gao, H., Ji, S.: Graph U-Nets. In: ICML, pp. 2083–2092. Macao, China (2019)
8. Glorot, X., Bengio., Y.: Understanding the difficulty of training deep feedforward neural networks. In: AISTATS, vol. 9, p. 249–256 (2010)

9. Gori, M., Monfardini, G., Scarselli, F.: A new model for learning in graph domains. In: IJCNN, pp. 729–734 (2005)
10. Guo, Z., Zhang, J., Liu, X., Hou, H., et al.: Neurometabolic characteristics in the anterior cingulate gyrus of Alzheimer's disease patients with depression: a 1h magnetic resonance spectroscopy study. BMC Psychiatr. **15**(1), 1–7 (2015)
11. Henaff, M., Bruna, J., LeCun, Y.: Deep convolutional networks on graph-structured data. arXiv preprint arXiv:1506.05163 (2015)
12. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: ICLR, San Diego, U.S. (2015)
13. Kipf, T., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR, Toulon, France (2017)
14. Klicpera, J., Weißenberger, S., Günnemann, S.: Diffusion improves graph learning. In: NIPS (2019)
15. McPherson, M., Smith-Lovin, L., Cook, J.: Birds of a feather: homophily in social networks. Ann. Rev. Sociol. **27**(1), 415–444 (2001)
16. Oppenheim, A., Schafer, R., Buck, J.: Signal and Systems, 2nd edn. Prentice Hall, Upper Saddle River, N.J. (1999)
17. Scheff, S.W., Price, D.A., Schmitt, F.A., Scheff, M.A., Mufson, E.J.: Synaptic loss in the inferior temporal gyrus in mild cognitive impairment and Alzheimer's disease. J. Alzheimers Dis. **24**(3), 547–557 (2011)
18. Scott, M.R., Hampton, O.L., Buckley, R.F., et al.: Inferior temporal tau is associated with accelerated prospective cortical thinning in clinically normal older adults. Neuroimage **220**, 116991 (2020)
19. Shuman, D., Ricaud, B., Vandergheynst, P.: Vertex-frequency analysis on graphs. ACHA **40**(2), 260–291 (2016)
20. Stam, C.J., de Haan, W., Daffertshofer, A., et al.: Graph theoretical analysis of magnetoencephalographic functional connectivity in Alzheimer's disease. Brain **132**(1), 213–224 (2008)
21. Tekin, S., Mega, M.S., Masterman, D.M., Chow, T., et al.: Orbitofrontal and anterior cingulate cortex neurofibrillary tangle burden is associated with agitation in Alzheimer disease. Ann. Neurol. **49**(3), 355–361 (2001)
22. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. In: ICLR, Vancouver, Canada (2018)
23. Venneri, A., McGeown, W.J., Hietanen, H.M., Guerrini, C., Ellis, A.W., Shanks, M.F.: The anatomical bases of semantic retrieval deficits in early Alzheimer's disease. Neuropsychologia **46**(2), 497–510 (2008)
24. Wu, T., Wang, L., Chen, Y., Zhao, C., Li, K., Chan, P.: Changes of functional connectivity of the motor network in the resting state in Parkinson's disease. Neurosci. Lett. **460**(1), 6–10 (2009)
25. Xu, B., Shen, H., Cao, Q., Cen, K., Cheng, X.: Graph convolutional networks using heat Kernel for semi-supervised learning. In: IJCAI, Macao, China (2019)
26. Zhang, M., Chen, Y.: Link prediction based on graph neural networks. In: NIPS (2018)