HIERARCHICAL BRAIN EMBEDDING USING EXPLAINABLE GRAPH LEARNING

Haoteng Tang[1], *Lei Guo*[1], *Xiyao Fu*[1], *Benjamin Qu*[2], *Paul M. Thompson*[3], *Heng Huang*[1], *Liang Zhan*[1,†]

1. Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, USA
2. Mission San Jose High School, Fremont, USA
3. Institute for Neuroimaging and Informatics, University of Southern California, Marina del Rey, USA

ABSTRACT

Brain networks have been extensively studied in neuroscience, to better understand human behavior, and to identify and characterize distributed brain abnormalities in neurological and psychiatric conditions. Several deep graph learning models have been proposed for brain network analysis, yet most current models lack interpretability, which makes it hard to gain any heuristic biological insights into the results. In this paper, we propose a new explainable graph learning model, named hierarchical brain embedding (HBE), to extract brain network representations based on the network community structure, yielding interpretable hierarchical patterns. We apply our new method to predict aggressivity, rule-breaking, and other standardized behavioral scores from functional brain networks derived using ICA from 1,000 young healthy subjects scanned by the Human Connectome Project. Our results show that the proposed HBE outperforms several state-of-the-art graph learning methods in predicting behavioral measures, and demonstrates similar hierarchical brain network patterns associated with clinical symptoms.

Index Terms— brain functional connectome, explainable AI, graph learning, regression, HCP

1. INTRODUCTION

Brain networks, derived from various non-invasive imaging techniques (such as diffusion MRI or resting state functional MRI), have been widely studied in diverse areas of neuroscience and clinical brain research [1, 2]. Although many studies have been conducted to predict behavioral, clinical, or psychiatric measures from brain networks, and identify the most predictive network features, most existing studies have focused on correlating clinical measures with a small number of pre-defined network features (e.g., [3]). This may be sub-optimal as the derived network features are often low-dimensional and may contain much less information than the original brain network. Using the entire brain network for this prediction task has also been extensively studied. Although some promising results (e.g., [4, 5]) have been achieved, how the information aggregates through the brain network and eventually links to the predicted target is not clear.

Recent years have witnessed enormous successes in deep learning. As a powerful tool to discover patterns in large-scale datasets, deep learning methods have also been widely applied to biomedical data to learn and find informative features that can describe the regularities inherent in medical data, as well as abnormalities in disease. For analysis of graph data (such as brain networks), graph learning techniques [6–8] have been gaining significant attention. An important issue for current graph learning methods is that the models are not typically easy to interpret. Many current graph learning methods may well achieve good predictive performance for some tasks (e.g., classification of disease or predictive modeling based on network data), but it might be difficult to provide any biological explanation or insight into the results.

In this work, we propose a new explainable graph representation learning model to predict behavioral and psychiatric measures using the entire brain network. We hypothesize that the whole brain network's intrinsic representation can be derived from graph communities within the brain network - in a hierarchical manner - and we hypothesize that this hierarchical pattern guides the information flow in our predictive models. Our proposed model explicitly uncovers the graph community partitions underlying different tasks (e.g., predicting different behavioral measures) and indicates the brain network's community partitions are quite similar for related predictive tasks.

2. METHODS

In this section, we first provide an overview of the proposed hierarchical brain network embedding (HBE) framework for a typical regression task. Then, we delve into the proposed graph pooling block which down-scales the graph and coarsens the graph representations based on the graph communities. Finally, we briefly describe the loss functions designed to train the proposed framework in an efficient, end-to-end manner.

2.1. Hierarchical Brain Network Embedding Framework

Let $G = (A, X)$ be any attributed brain network with N nodes, where $A \in \mathcal{R}^{N \times N}$ is the graph adjacency matrix.

† Corresponding Author

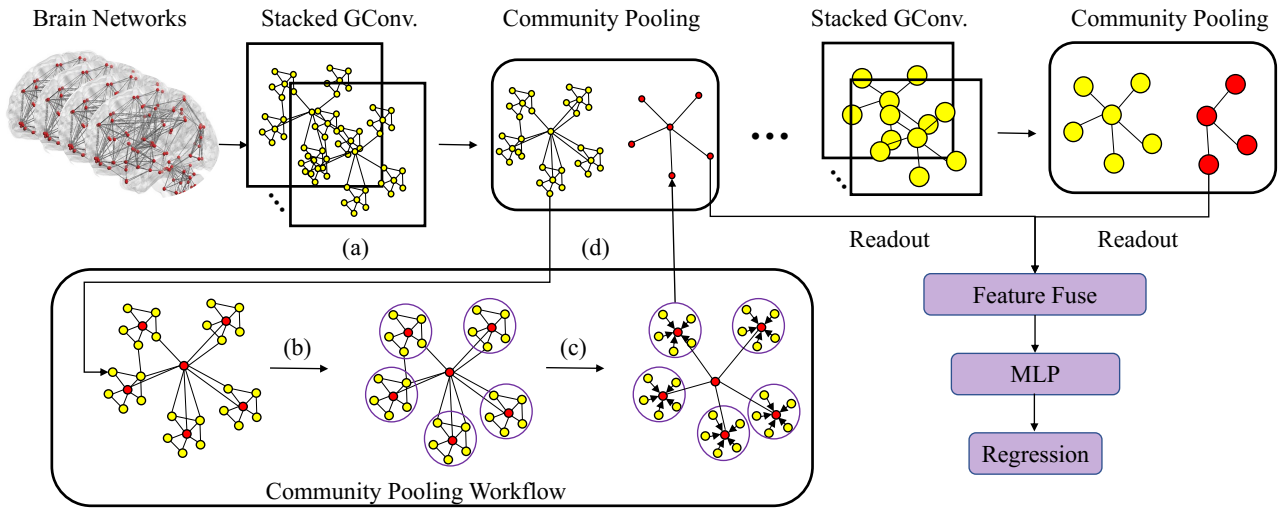

Fig. 1. Diagram of the proposed hierarchical brain network learning framework, including a stacked graph convolution block (Stacked GConv.), a Community-based Pooling block, and the Multilayer perceptron (MLP) block for the regression task. The operations performed by these blocks are to: (a) Compute the centroid node probability ($\mathcal{P}$) and select the nodes with top-M $\mathcal{P}$ scores as centroid nodes. (b) Assign each node into the closest community. (c) Aggregate features of community member nodes to the corresponding centroid node. (d) Down scale the graph based on the communities.

$X \in \mathcal{R}^{N\times d}$ is the node feature matrix, where the feature dimension is d. We use $Z = [Z_1, ..., Z_N] \in \mathcal{R}^{N\times c}$ to denote the latent features of N nodes embedded by the graph convolution layers, where c is the dimension of the node latent features. As shown in Figure 1, the proposed hierarchical brain network learning framework consists of three components: a node embedding block, a community-based graph pooling block, and the task-specific prediction block. In the node embedding block, we deploy stacked graph convolution layers which can enable each graph node to aggregate higher order information from several-hops neighborhoods[9]. Following[10], each graph convolution layer can be formulated as:

$$Z = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}X\theta_1) \tag{1}$$

where $\tilde{A} = A + I$, $\tilde{D}_{ii} = \sum_{:,j}\tilde{A}_{i,j}$ is the degree matrix, θ_1 is a trainable parameters and $\sigma(\cdot)$ is a nonlinear activation function.

The goal for the graph pooling block is to down-scale the graph from N nodes to the $M(< N)$ nodes based on the graph communities. After the graph pooling block, the graph latent features $Z \in \mathcal{R}^{N\times c}$ will be down-scaled to $\bar{Z} = [\bar{Z}_1, ..., \bar{Z}_M] \in \mathcal{R}^{M\times c}$. Details of the community-based graph pooling block are presented in the next subsection.

Note that each graph pooling block is followed by a readout operation which is used to summarize the whole graph representation at the current scale of graph. Given a graph latent feature matrix $\bar{Z}$ down-scaled by a graph pooling layer, the readout function summarizes the whole graph representation (i.e., Z_G) by summing $\bar{Z}_i$, where $i \in \{1, ..., M\}$.

After the last pooling block, we fuse (i.e., concatenate) all the Z_G obtained under different scales of graphs as the hierarchical graph representation for the final prediction task. In the prediction block, we deploy Multilayers Perceptron (MLP) to transform the fused Z_G for the graph regression task.

2.2. Community-Based Graph Pooling Block

As mentioned already, the graph pooling block takes the node latent features $Z \in \mathcal{R}^{N\times c}$ as the input and generates the down-scaled node feature matrix $\bar{Z} \in \mathcal{R}^{M\times c}$ based on the community structures. Therefore, the most important step in this pooling step is to identify the community centroid nodes and assign other nodes to the nearest community based on node features. From the viewpoint of density-based clustering methods [11], a community centroid node is densely encircled by a group of nodes with a high probability. Inspired by this, we use the feature distances as a metric to approximate the probability that a given node feature indicates that the corresponding node is the centroid node. In other words, a node with smaller feature distances to all other nodes will have a higher chance of being a community centroid node. Therefore, we create a feature distance matrix S where $S_{i,j} = \|Z_i - Z_j\|_{L_1}$ to measure the density among node features. Based on the matrix S, we compute the probability vector ($\mathcal{P} \in \mathcal{R}^{N\times 1}$) for each node as a community centroid node, by:

$$\mathcal{P} = softmax(\vec{1} - normalize[\sum_{j=1}^{N} S_{i,j}]) \tag{2}$$

where the normalize function maps the feature distances into $[0, 1]$ as probabilities. Finally, we select the M nodes with

Top-M $\mathcal{P}$ values as M community centroid nodes.

After we determine M community centroid nodes, we assign other graph nodes into the closest community to generate M community partitions (i.e., where $\Omega = \{\Omega_1, ..., \Omega_M\}$ represents the set of all communities). Then the community representation (e.g., $\bar{Z}_i$ for community-i, $i \in \{1, ..., M\}$) can be computed by:

$$\bar{Z}_i = Z_{c_i} + \sum_{v_j \in \Omega_i} Z_{v_j} \cdot \frac{1}{S_{j,i}} \tag{3}$$

where Z_{c_i} is the latent feature of the centroid node of community-i. v_j are the community member nodes in the community.

2.3. Loss Functions

First, we optimize ℓ_{MSE} to minimize the difference between model output $\hat{y}$ and the ground-truth y. Meanwhile, we encourage the feature of community members to be close to the corresponding community centroid by minimizing:

$$\ell_{KL} = \sum_{\Omega_i \in \Omega} \sum_{v_j \in \Omega_i} KL(\tilde{Z}_{v_j} \| \tilde{Z}_{c_i}) \tag{4}$$

where $\tilde{Z}_{v_j}$ and $\tilde{Z}_{c_i}$ are normalized as probability distributions and KL is the Kullback–Leibler loss. The total loss function can be formulated as follows:

$$L_{reg} = \ell_{MSE}(\hat{y}, y) + \ell_{KL} \tag{5}$$

3. RESULTS AND DISCUSSIONS

3.1. Data Description and Implementation Details

Our experimental data was downloaded from the publicly available Human Connectome Project dataset [12] (HCP) and contains neuroimaging data from 1,000 young healthy subjects (mean age=28.84 $\pm$ 3.69, 544 women). Each subject has a brain network representation of dimension of 50×50 , derived from resting-state functional MRI using the ICA (independent components analysis) method. The details of network reconstruction pipeline can be found in the HCP official website[1]. The adjacent matrix of each subject is computed by the absolute value of the resting-state functional network. For each node, the nodal features include min, 25%, median, 75%, max of BOLD signal at that node. We select three standardized clinical measures (age and sex-adjusted aggressivity score, intrusiveness score, and rule-breaking score, from the Adult Self-Report scale, or ASR) as targets for our prediction tasks. These scores are widely-used behavioral measures that we aim to predict from our network representations. The details of these three ASR scores can be found at the HCP official website.

[1] https://wiki.humanconnectome.org

	Aggr.	Rule	Intr.	Overall
PCA + LR	2.97(530)	3.49(12)	3.77(2.6)	6.26(27)
Spec. C + LR [15]	2.64(66)	2.53(2.4)	2.10(4.2)	5.08(33)
Global Pool	3.24(760)	2.86(130)	2.42(330)	5.97(10)
SAG Pool[13]	1.24(240)	1.66(270)	1.25(4.8)	4.07(8.2)
DIFFPOOL[16]	1.79(110)	1.58(4.8)	1.17(71)	3.72(47)
HGP-SL[17]	1.11(19)	1.24(26)	1.21(2.6)	3.16(110)
StructPool [18]	1.57(21)	1.11(720)	1.36(19)	2.94(6.2)
HBE w/o KL†	**1.02(5.2)**	**0.87(113)**	**1.21(36)**	**2.17(12)**
HBE	**0.82(66)**	**0.71(125)**	**1.02(12)**	**1.98(24)**

Table 1. Regression Mean Absolute Error (MAE) with corresponding standard deviations ($\times \mathbf{10^{-5}}$) under 5-fold cross-validation. The values in **bold** show the best and second best results. † shows the results of our model without using KL loss to optimize the community inner features. LR and Spec. C are Linear Regression and Spectral Clustering respectively. Overall denotes the task of jointly predicting all three scores.

We randomly split the entire dataset into 5 disjoint sets for 5-fold cross-validations in the following experiments. All the hyperparameters (e.g., initial learning rate, dimension of the latent features, pooling ratios etc.) are same across each validation experiment. We trained the model using the Adam optimizer with a batch size of 128. The initial learning rate was set to 0.001 and decayed by $(1 - \frac{current\_epoch}{max\_epoch})^{0.9}$. We also regularized the training with an L_2 weight decay of $1e^{-5}$. We stopped the training if the validation loss did not improve for 40 epochs in an epoch termination condition with a maximum of 500 epochs, as was done in [13, 14]. The experiments were deployed on one NVIDIA TITAN RTX GPU.

3.2. Regression Performance

We compare the proposed Hierarchical Brain Embedding (HBE) model with 7 baselines to show the superiority of our model. The baselines include 2 dimension reduction methods (i.e., PCA and Spectral Clustering with linear regression) and 5 graph neural network models with different pooling strategies. The 2 dimension reduction methods and Global Pool maintain the number of graph nodes and we average all 50 node features for the final prediction. For the 4 hierarchical graph pooling baselines (i.e., SAG Pool, DIFFPOOL, HGP-SL and StructPool) and our HBE model, we deploy two Embedding and Pooling (E-P) blocks and the pooling ratio is set to 0.4, which will scale down the number of brain nodes from 50 to 20 and then to 8 in a hierarchical manner. Table 1 shows that our methods outperform all the baselines in predicting each ASR score (Aggressivity, Rule-breaking and Intrusiveness) as well as jointly predicting all three ASR scores. The best results were achieved by using KL loss to make inner community features closer. In general, the hierarchical pooling strategies performed better than other methods, indicating that the hierarchical graph structures are important for the whole graph representation.

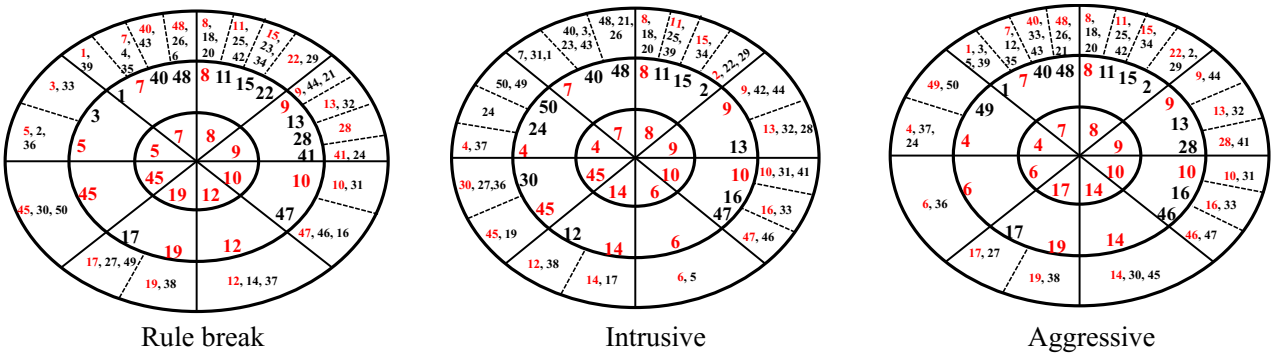

Fig. 2. Community-based hierarchical pooling derived from the HBE model. The red numbers indicate community centroid nodes. Each small patch in circles indicates a community. The graph is scaled down from the outer to the inner circle.

3.3. Statistical Analysis and Ablation Studies

(a). To evaluate the significance of the prediction performance of our HBE model, we design a permutation test by randomly selecting different sets of 20 nodes from 50 nodes and randomly select different 8 nodes from 20 nodes in the first and second pooling blocks respectively for the prediction. This process was repeated 10^4 times on the aforementioned four prediction tasks and we rank the original prediction accuracy among these 10^4 permutation tests. Our results show that the results from the proposed HBE model are significant in these four tasks (P values are 7×10^{-4}, 1×10^{-4}, 3×10^{-4}, and 11×10^{-4} respectively).

(b). To show that the hierarchical pooling operation in the HBE model is necessary and beyond a dimension reduction strategy, we directly select the 8 key nodes, identified by the last pooling layer of HBE, and adopt a stacked graph convolution block to generate these 8 key nodes' embedded features and use them to train the MLP block for regression tasks. Table 2 shows that the prediction performances, when using the 8 key nodes, are much worse than those of HBE for all tasks, which may due to that the HBE model summarizes not only the information of each community member but also the local structures onto the corresponding community centroid node, while this kind of semantic information is ignored when only embedding 8 nodes.

(c). To show that our results are stable under different hyperparameters, we present the regression results with different (1) numbers of E-P blocks, (2) pooling ratios and (3) dimensions of initialized node features for jointly predicting three ASR scores. First, we fix the pooling ratio and initialized feature dimensions as 0.4 and 4 respectively, the performance of HBE when the number of E-P blocks ranges in $[1, 2, 3]$ are $[2.17(7.8\times10^{-5}, 2.01(1.3\times10^{-3}), 2.06(2.6\times10^{-4})]$. Second, we fix the number of E-P blocks and pooling ratio as 2 and 0.4 respectively, the performance of HBE when the initialized feature dimension ranges in $[2, 4, 6]$ are $[2.09(3.0\times10^{-4}), 2.01(1.3\times10^{-3}), 2.17(9.8\times10{-}5)]$. Finally, we fix the number of E-P blocks and initialized feature dimension as 2 and 4 respectively, the performance of HBE when the pooling ratio ranges in $[0.3, 0.4, 0.5]$ are $[2.32(8.1\times10^{-5}), 2.01(1.3\times10^{-3}), 2.22(1.9\times10^{-4})]$. All these results are very similar to our result $(1.98(2.4\times10^{-4}))$ reported in the last column of Table 1, which demonstrates the stability of our HBE model. Note that the results in parameter ablation studies are different from the Table 1. The reason is that the initialized features of graph nodes here are randomly sampled from a Gaussian distribution, however, the initialized graph node features in Table 1 and 2 are set as 25%, median, 75%, max of BOLD signal, which is mentioned in section 3.1.

	Aggr.	Rule	Intr.	All
8-nodes	2.25(27)	3.07(79)	3.91(102)	7.62(11)
HBE	**0.82(66)**	**0.71(125)**	**1.02(12)**	**1.98(24)**

Table 2. Regression Mean Absolute Error (MAE) with corresponding standard deviation ($\times\mathbf{10^{-5}}$) under 5-fold cross-validation. The values in **bold** show the best results.

3.4. Visualization of HBE patterns

Figure 2 illustrates how the information is hierarchically aggregated from the entire brain network to nodes and eventually can be used in the regression model to predict the clinical symptoms, which indicates similar cooperativity among nodes in predicting similar behavioral scores. For example, node 7, 8, 9 and 10 always serve as community centers and cooperate together in the whole graph representation among the three tasks. Future work will be conducted to explore the biological basis for these hierarchical patterns.

4. CONCLUSION

Here we propose a new explainable hierarchical graph learning framework, HBE, to capture the graph representations based on the community structures. We deploy the proposed framework to learn the patterns of brain networks for predicting three behavioral scores. Our experimental results demonstrate the superiority of our HBE model, compared to various baseline methods. Meanwhile, the proposed HBE model explicitly uncovers the informative graph hierarchical patterns' similarities across three related tasks.

5. ACKNOWLEDGEMENT

This study is partially supported by the National Institutes of Health (R01AG071243, R01MH125928 and U01AG068057) and National Science Foundation (IIS 2045848 and IIS 1837956).

Data were provided by the Human Connectome Project, MGH-USC Consortium (Principal Investigators: Bruce R. Rosen, Arthur W. Toga and Van Wedeen; U01MH093765) funded by the NIH Blueprint Initiative for Neuroscience Research grant; the National Institutes of Health grant P41EB01-5896, and the Instrumentation Grants S10RR023043, 1S10RR-023401, 1S10RR019307.

Part of the work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1548562. Specifically, it used the Bridges system, which is supported by NSF award number ACI-1445606, at the Pittsburgh Supercomputing Center (PSC).

6. COMPLIANCE WITH ETHICAL STANDARDS

The research study was conducted retrospectively using human subject data made available in open access by Human Connectome Project, MGH-USC Consortium. The details of data source used in the paper has been provided in the Section 3. Ethical approval was not required as confirmed by the license attached with the open access data.

References

[1] Laura E Korthauer, Liang Zhan, Olusola Ajilore, A Leow, and Ira Driscoll, "Disrupted topology of the resting state structural connectome in middle-aged apoe ε4 carriers," *Neuroimage*, vol. 178, pp. 295–305, 2018.

[2] Yanfu Zhang, Liang Zhan, Weidong Cai, Paul Thompson, and Heng Huang, "Integrating heterogeneous brain networks for predicting brain disease conditions," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 214–222.

[3] Rosaleena Mohanty, William A Sethares, Veena A Nair, and Vivek Prabhakaran, "Rethinking measures of functional connectivity via feature extraction," *Scientific reports*, vol. 10, no. 1, pp. 1–17, 2020.

[4] Xingjuan Li, Yu Li, and Xue Li, "Predicting clinical outcomes of alzheimer's disease from complex brain networks," in *ADMA*. Springer, 2017, pp. 519–525.

[5] Yurong Chen, Haoteng Tang, Lei Guo, Jamie C Peven, Heng Huang, Alex D Leow, Melissa Lamar, and Liang Zhan, "A generalized framework of pathlength associated community estimation for brain structural network," in *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2020, pp. 288–291.

[6] William L Hamilton, Rex Ying, and Jure Leskovec, "Inductive representation learning on large graphs," in *NeurIPS*, 2017, pp. 1025–1035.

[7] Haoteng Tang, Guixiang Ma, Lifang He, Heng Huang, and Liang Zhan, "Commpool: An interpretable graph pooling framework for hierarchical graph representation learning," *Neural Networks*, vol. 143, pp. 669–677, 2021.

[8] Cătălina Cangea, Petar Veličković, Nikola Jovanović, Thomas Kipf, and Pietro Liò, "Towards sparse hierarchical graph classifiers," *arXiv preprint arXiv:1811.01287*, 2018.

[9] Nima Dehmamy, Albert-László Barabási, and Rose Yu, "Understanding the representation power of graph neural networks in learning graph topology," *NeurIPS*, vol. 32, pp. 15413–15423, 2019.

[10] Thomas N Kipf and Max Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[11] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al., "A density-based algorithm for discovering clusters in large spatial databases with noise.," in *KDD*, 1996, vol. 96, pp. 226–231.

[12] David C Van Essen, Stephen M Smith, Deanna M Barch, Timothy EJ Behrens, Essa Yacoub, Kamil Ugurbil, Wu-Minn HCP Consortium, et al., "The wu-minn human connectome project: an overview," *Neuroimage*, vol. 80, pp. 62–79, 2013.

[13] Junhyun Lee, Inyeop Lee, and Jaewoo Kang, "Self-attention graph pooling," in *ICML*. PMLR, 2019, pp. 3734–3743.

[14] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann, "Pitfalls of graph neural network evaluation," *arXiv preprint arXiv:1811.05868*, 2018.

[15] Andrew Y Ng, Michael I Jordan, and Yair Weiss, "On spectral clustering: Analysis and an algorithm," in *NeurIPS*, 2002, pp. 849–856.

[16] Rex Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L Hamilton, and Jure Leskovec, "Hierarchical graph representation learning with differentiable pooling," in *NeurIPS*, 2018, pp. 4805–4815.

[17] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang, "Hierarchical graph pooling with structure learning," *AAAI*, 2020.

[18] Hao Yuan and Shuiwang Ji, "Structpool: Structured graph pooling via conditional random fields," in *ICLR*, 2020.