

# TRANSFORMER-BASED HIERARCHICAL CLUSTERING FOR BRAIN NETWORK ANALYSIS

Wei Dai<sup>†</sup> Hejie Cui<sup>‡</sup> Xuan Kan<sup>‡</sup> Ying Guo<sup>†</sup> Sanne van Rooij<sup>‡</sup> Carl Yang<sup>†\*</sup>

<sup>†</sup> Stanford University, <sup>‡</sup> Emory University

## ABSTRACT

Brain networks, graphical models such as those constructed from MRI, have been widely used in pathological prediction and analysis of brain functions. Within the complex brain system, differences in neuronal connection strengths parcellate the brain into various functional modules (network communities), which are critical for brain analysis. However, identifying such communities within the brain has been a nontrivial issue due to the complexity of neuronal interactions. In this work, we propose a novel interpretable transformer-based model for joint hierarchical cluster identification and brain network classification. Extensive experimental results on real-world brain network datasets show that with the help of hierarchical clustering, the model achieves increased accuracy and reduced runtime complexity while providing plausible insight into the functional organization of brain regions. The implementation is available at <https://github.com/DDVD233/THC>.

**Index Terms**— Brain Networks, Neural Imaging Analysis, Graph Neural Networks, Clustering, Machine Learning

## 1. INTRODUCTION

Graph is a ubiquitous form of data as it captures multiple objects and their interactions simultaneously. It is widely used for representing complex systems of related entities [1, 2]. Brain network is a special kind of graph constructed from MRI images. In brain networks, the anatomical areas named “Region of Interests” (ROIs) are represented as nodes, while connectivities between ROIs are represented as links. Partition atlas defines the set of ROIs in a particular brain network. In recent decades, abundant works have shown strong associations between linking imaging-based brain connectivity and demographic characteristics or mental disorders [3–7].

Both shallow machine learning models like M2E [8] and deep models like graph neural networks (GNNs) [9] are researched in the area of brain network analysis. Shallow models exhibit inferior performance as compared to deep models [3], but GNNs also suffer from over-smoothing [10], which limits their ability to model long-distance interactions. Transformers, on the other hand, have recently emerged as

a promising approach for various tasks [11, 12], including predictions on graph data [13]. Graph based transformers utilize pairwise attention across full graphs, unlike GNNs, which only propagate node embeddings locally. BrainTransformer [14] employs transformer on brain networks and demonstrates state-of-the-art performance for brain network analysis.

ROIs in Brain networks are inherently hierarchically clustered [15]. In typical brain network analysis [16], clustered ROIs form communities, with each representing a particular functional module. These functional modules are then further organized into larger functional modules, with each responsible for a more general function. This arrangement creates a hierarchical “module-in-module” structure [17]. Functional modules provide critical information with regard to downstream tasks, and alterations in community patterns sometimes signal pathological lesions [18]. Therefore, learning a globally shared cluster assignment with the awareness of downstream tasks is beneficial for both model optimization and clinical examinations. However, learning such hierarchical cluster representations is difficult. Shallow methods proposed to detect brain communities are mostly based on the Louvain algorithm [19] and Lloyd algorithm [20]. These correlation-based methods fail to capture higher-order connectivity patterns between brain regions [21]. Some GNN models are proposed to detect communities within the brain [9], but these models suffer from the over-smoothing problem of GNN, limiting their ability to aggregate and identify multi-hop connectivity patterns.

To solve the aforementioned challenges, we propose a Transformer-based Hierarchical Clustering model, abbreviated as THC, that is tailored for brain network analysis. We highlight three main contributions of our clustering model. First, we offer an end-to-end transformer based approach to learn clustering assignments. Through pairwise attention, a clustering layer, BCluster, and a transformer encoder collaboratively learn a globally shared clustering assignment that is continuously tuned to downstream tasks. BCluster enhances the model’s performance, reduces run time complexity, while also providing clinical insights. Second, we propose a hierarchical structure for the clustering model, enabling the model to learn more abstract, higher-level cluster representations by combining lower-level modules. Each clustering layer

\*Corresponding author: Carl Yang <j.carlyang@emory.edu>

is attached to a distinct readout module, which allows the model to effectively utilize the cluster embeddings of every layer. Last but not least, we redesign the attention mechanism of the transformer with stochastic noise, which enhances its cluster learning capability. We compare our model’s performance with SOTA models and perform clustering analysis with the ground truth community labels [22]. Empirical analysis demonstrates the superior prediction power of our model, and the assignment produced by THC aligns well with the ground-truth functional module labels.

## 2. RELATED WORKS

**Deep Learning for Brain Network Analysis.** Deep learning models have gained widespread application in brain network analysis due to their ability to discover shared patterns across samples with diverse characteristics [23]. To this end, IBGNN [24] has emerged as one of the first interpretable graph neural networks (GNNs) for brain networks. Graph-based transformers such as SAN [13] and Graphormer [25] have been recently introduced as promising alternatives to GNN. Brain Network Transformer [14] presents a novel approach that includes unique readout functions and optimal positional embeddings and attention mechanisms for brain networks. BrainNN [26] and CroGen [27] explore the integration of multiview brain networks. However, despite the promise of the existing models, they lack the ability to provide the clustering information that is essential for brain network analysis. **Cluster-aware GNNs.** Brain networks have overlapping communities [28]. To cluster nodes, DiffPool [29] uses an auxiliary GNN, GRACE [30] utilizes stable states of repeated propagation, whereas HoscPool [31] uses probabilistic motif spectral clustering. GNN models like BrainGNN [9] detect communities, but over-smoothing limits their ability to identify complex cluster structures [32].

## 3. TRANSFORMER-BASED HIERARCHICAL CLUSTERING (THC)

**Problem Definition** Our model’s input is a weighted adjacency matrix  $\mathbf{X} \in \mathbb{R}^{V \times V}$  of a brain network, where  $V$  is the number of nodes (ROIs) as defined in the network. The objective of the model is to predict the sample class  $y$  and a  $k$ -layer hierarchical cluster assignment  $(\mathbf{A}^1, \dots, \mathbf{A}^k)$ . As shown in Figure 1, the overall model can be broken down into three modules: a Stochastic Transformer (StoTrans) encoder, a Brain Clustering (BCluster) module and a Readout.

**Transformer Based Clustering.** The stacked BCluster layer learns globally shared cluster assignments  $(\mathbf{A}^1, \dots, \mathbf{A}^k)$ , where  $k$  is the number of model layers. Each assignment  $\mathbf{A}^i \in (C_i, C_{i+1})$  reduces  $\mathbf{X}^i \in (C_i, C_i)$  into  $\mathbf{X}^{i+1} \in (C_{i+1}, C_{i+1})$ , where  $C_{i+1} \in (C_1, \dots, C_k)$  are the *hyper-parameters* denoting the cluster sizes of the  $i^{th}$  layer. Specifically, at layer  $i$ , the adjacency matrix  $\mathbf{X}^i$  is first fed into the

transformer encoder, which learns attention matrix  $\mathbf{S}^{(i,m)}$

$$\mathbf{S}^{(i,m)} = \frac{\mathbf{W}_{\mathcal{Q}}^{i,m} \mathbf{X}^i (\mathbf{W}_{\mathcal{K}}^{i,m} \mathbf{X}^i)^\top}{\sqrt{d_{\mathcal{K}}^{i,m}}} \quad (1)$$

where  $\mathbf{W}_{\mathcal{Q}}^{i,m}$  and  $\mathbf{W}_{\mathcal{K}}^{i,m}$  are learnable parameters. A node embedding  $\mathbf{X}^{i'}$  is also learned through vanilla transformer propagation using the attention matrix [33].

$$\mathbf{X}^{i'} = \frac{1}{M} \sum_{m=1}^M \text{Softmax}(\mathbf{S}^{(i,m)}) \mathbf{X}^i \mathbf{W}_{\mathcal{V}}^i \quad (2)$$

where  $\mathbf{W}_{\mathcal{V}}^i$  is a learnable parameter. Then, a batch-wise shared clustering assignment is obtained by

$$\mathbf{A}^i = \text{Softmax} \left( \frac{1}{M} \sum_{m=1}^M \mathbf{S}^{(i,m)} \mathbf{W}_{\mathcal{A}}^i \right) \quad (3)$$

where  $\mathbf{W}_{\mathcal{A}}^i$  is a learnable parameter and the Softmax is applied row-wise. The learned node embedding  $\mathbf{X}^{i'}$  is fed into an encoder MLP layer, followed by the multiplication of the globally shared clustering assignment  $\mathbf{A}^i$ :

$$\mathbf{X}^{i+1} = \text{MLP}(\mathbf{X}^{i'}) \cdot \mathbf{A}^i. \quad (4)$$

The main objective of BCluster is maximizing the mutual information between the pre-clustered embedding  $\mathbf{X}^{i'}$  and cluster embedding  $\mathbf{X}^{i+1}$ . This objective encourages the model to cluster similar nodes as the information loss of combining nodes with similar representations is also minimal. Inspired by [24], we supervise the parameter  $\mathbf{W}_{\mathcal{A}}$  jointly with the full model through cross-entropy loss. Two losses, sparsity loss and element-wise entropy loss are added to regularize the number of clusters each node is assigned to

$$\mathcal{L}_{\text{SPS}} = \sum_{i,j} \mathbf{A}_{i,j}, \quad (5)$$

$$\mathcal{L}_{\text{ENT}} = -(\mathbf{S} \log(\mathbf{A}) + (1 - \mathbf{A}) \log(1 - \mathbf{A})). \quad (6)$$

The final objective is the sum of all three losses. The BCluster layer has the following favorable properties:

1. *The Assignment is Globally Shared.* During the training process, the assignment is shared batch-wise and optimized jointly with the model. After training is complete, a final globally shared assignment is obtained by averaging across the assignments of all samples.
2. *The Assignment is Differentiable.* Each node/cluster  $\mathcal{V}^{i,j} \in \mathbf{X}^i$  at layer  $i$  is assigned a vector of probabilities  $(p_{\mathcal{V}^{i,j},1}, \dots, p_{\mathcal{V}^{i,j},k})$ , with  $p_{\mathcal{V}^{i,j},k}$  denoting the probability of assigning  $\mathcal{V}^{i,j}$  to  $k^{th}$  cluster. This soft assignment is differentiable, allowing the model to optimize the assignment directly through gradient descent.
3. *Clustering Reduces Computational Complexity.* At  $i^{th}$  layer, the computational complexity of each transformer encoder is  $O(n^2d + nd^2)$ , where  $n$  represents the number of nodes and  $d$  represents the number of features. The clustering reduces the complexity to  $O(k^2d + kd^2)$ , where  $k$  is the cluster count and  $k \ll n$ .

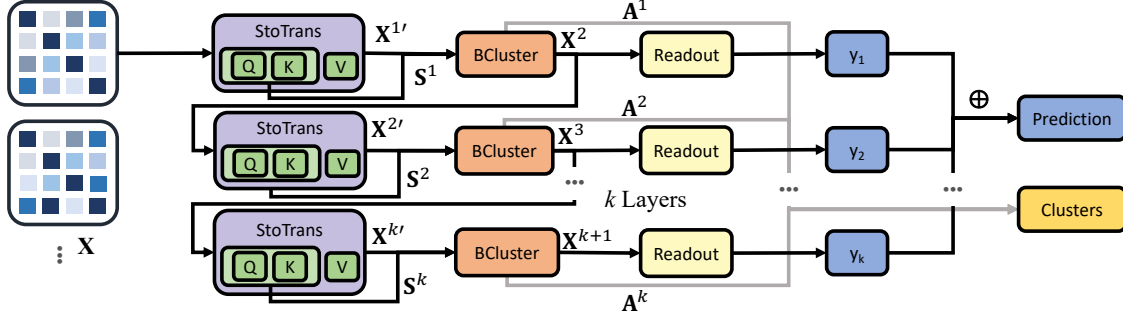


Fig. 1: The overall framework of our proposed model THC.

4. *The Assignment is Squashable.* A hierarchical clustering assignment produced by BCluster can be flattened into one-layer clustering with a linear number of matrix multiplications. The  $k$ -layer hierarchical assignments  $(\mathbf{A}^1, \dots, \mathbf{A}^k)$  are reduced into one assignment  $\mathbf{A}^{flat}$  by multiplication,  $\mathbf{A}^{flat} = \prod_{i=1}^k \mathbf{A}^i$ .

**Hierarchical Structure.** The model further groups the cluster embeddings from the previous layer into new, larger clusters, generating a hierarchical structure. A readout layer is attached to each BCluster layer. At layer  $i$ , an MLP Readout is attached to the BCluster layer of that layer  $\mathbf{X}^i$  and outputs a prediction  $y_i$ . This allows the model to effectively ensemble cluster embeddings from different layers. The final prediction  $y$  is the average of all outputs of each Readout layer, where  $y = \frac{1}{k} \sum_{i=1}^k y_i$ .

**Stochastic Noise.** To allow BCluster to effectively learn the clustering assignment without falling into the trivial situation of replicating the results from the attention matrix, we redesign attention matrix  $\mathbf{S}^{(i,m)}$  with stochastic noise

$$\mathbf{S}^{(i,m)'} = \mathbf{S}^{(i,m)} + \log\left(\frac{\mathbf{B}^{i,m}}{\mathbf{I} - \mathbf{B}^{i,m}}\right) \quad (7)$$

where  $\mathbf{B} \sim \mathbf{S}^{i,m}$  is a random variable chosen uniformly from range  $(0, 1)$  and  $\mathbf{I} \sim \mathbf{S}^{i,m}$  is a diagonal matrix of ones.

## 4. EXPERIMENTS

**Datasets.** The method is tested on two fMRI datasets.

1. *Adolescent Brain Cognitive Development Study (ABCD):* This research enrolls children aged 9 to 10 years old. Repeated MRI scans are used to track each child until early adulthood [34]. Each sample is labeled with its biological sex. 7,901 children are involved in the study, with 3,961 (50.1%) of them being female. The parcellation is based on HCP 360 atlas [35].
2. *Autism Brain Imaging Data Exchange (ABIDE):* This dataset collects resting-state functional magnetic resonance imaging (rs-fMRI) [36]. It contains brain networks from 1009 subjects, with 516 (51.14%) being labeled as Autism spectrum disorder (ASD) patients (positives). The region is defined based on Craddock 200 atlas [37].

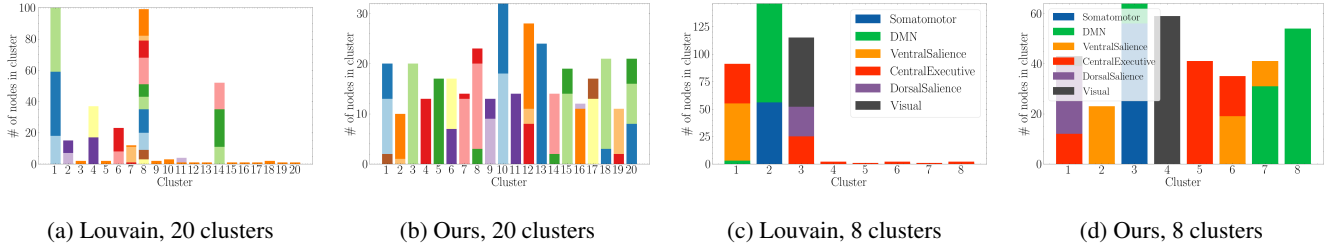
Table 1: Performance comparison of our model with different baselines in terms of AUROC and accuracy.

Type	Method	Dataset: ABIDE		Dataset: ABCD	
		AUROC	Accuracy	AUROC	Accuracy
Graph Transformer	SAN	71.3±2.1	65.3±2.9	90.1±1.2	81.0±1.3
	VanillaTF	76.4±1.2	65.2±1.2	94.3±0.7	85.9±1.4
	Graphormer	63.5±3.7	60.8±2.7	89.0±1.4	80.2±1.3
Fixed Network	BrainGNN	62.4±3.5	59.4±2.3	OOM	OOM
	BrainGB	69.7±3.3	63.6±1.9	91.9±0.3	83.1±0.5
	BrainNetCNN	74.9±2.4	67.8±2.7	93.5±0.3	85.7±0.8
Pooling & Clustering	DiffPool	68.3±0.9	61.3±1.7	83.2±1.2	75.5±1.4
	THC + Lin. Cluster	71.8±1.1	64.8±1.3	91.7±0.5	82.6±1.8
	THC + No Cluster	75.6±1.2	68.0±1.4	94.5±0.7	86.4±1.1
Ours	THC Full	<b>79.76±1.1</b>	<b>70.6±2.2</b>	<b>96.2±0.5</b>	<b>89.4±0.4</b>

**Baselines and Experiment Setup.** We compare the THC model with two kinds of baselines, as demonstrated in Table 1. Specifically, SAN [13], Graphormer [25] and Vanilla Transformer are transformer-based baselines, with the first two designed for graph prediction, and the vanilla transformer is a two-layer transformer based on the original implementation [33]. Three methods optimized for brain networks, BrainGNN [9], BrainGB [3], and BrainNetCNN [38], are also included. Our model is experimented with two layers and clustering sizes 90, 4 and 20, 4 for ABCD and ABIDE dataset respectively. We split the dataset randomly into a training set, a test set and a validation set on a 7:2:1 ratio. AUROC and accuracy are used over a 5-fold run as metrics. For all methods, the test performance on the epoch with the highest average AUROC on the validation set is recorded.

**Performance Analysis.** Performance results are recorded in Table 1. Compared with the baseline methods, it is shown that our model achieves the best performance on both datasets, with up to 3.2% and 1.7% higher AUROC on ABIDE and ABCD datasets over all transformer and GNN-powered models. Out of the 7 baselines, BrainGNN is the only model that generates community clustering. Nevertheless, its need of memory prevents its application on large graphs like ABCD.

To understand how our clustering method affects the model performance, we designed two ablative variants of our model: one with the clustering layer removed (THC + No Cluster), and one with BCluster swapped with a simple linear layer (THC + Lin. Cluster). We further compare the



**Fig. 2:** Hierarchical 2-layer cluster assignments with 20 and 8 clusters produced by THC and Louvain. In each cluster, nodes are colored in accordance with their ground-truth cluster labels. The legend for 20-cluster plots is omitted for simplicity.

performance of our model with a popular pooling model, DiffPool [29]. The results are shown in the Pooling & Clustering section of Table 1. Compared with the no clustering model, we observe that our full model gives a significant performance boost of up to 1.7% in AUROC and 3.0% in accuracy. Furthermore, we discover that a naive linear clustering negatively affects performance, proving that clustering assignments cannot be modeled in a simple way.

**Table 2:** Runtime comparison of our model with different baselines (in minutes) averaged over 5 runs.

Dataset	SAN	Graphormer	Vanilla TF	<b>THC</b>
ABCD	908.05±3.6	4089.86±5.7	36.26±2.12	<b>27.31±0.47</b>
ABIDE	93.01±0.96	133.52±0.54	2.32±0.10	<b>1.81±0.03</b>

**Runtime Analysis.** We compare the run time of our model with transformer baselines on an RTX 8000 GPU and record the results in Table 2. Our model runs faster than all baselines with a margin of at least 24.7% on ABCD and 22.0% on ABIDE, which demonstrate that our model can effectively reduce runtime complexity through the efficient aggregation of node embeddings into cluster embeddings of smaller sizes.

## 5. CLUSTERING ANALYSIS

To investigate the quality of the model’s clustering assignment, we run our model with 2 layers of cluster size 20 and 8, and create a bar plot of each cluster, as shown in Figure 2. The nodes within the 20 clusters assignment and 8 clusters assignments, represented by blocks in the cluster bar, are colored with respect to the AAc architecture at 20 modules and 6 modules definition by [22] respectively. In these bar plots, cluster purity is represented by the percentage of the majority color of that cluster. The results reveal that the proposed clustering method is able to produce clustering assignments similar to the existent labels for all functional modules. Among all the plotted clusters, cluster 2, 3, 4, 6 and 8 have purities of over 90 percent. Similar patterns are observed in the 20 clusters plot. Misclassified samples also provide interesting insights. In the 8 clusters sample, some Ventral Saliency Network nodes are misclassified as Default Mode Network

(DMN). This is supported by studies showing some areas of these two communities share similar functions [39].

**Table 3:** Cluster quality comparison of our model with baselines. Clustering results are compared with the labels by [22].

Method	Purity	NMI	Homogeneity
Lloyd	0.842±0.003	0.738±0.001	0.711±0.002
Louvain	0.671±0.099	0.647±0.052	0.648±0.087
GRACE	0.717±0.067	0.685±0.022	0.722±0.053
<b>THC</b>	<b>0.889±0.008</b>	<b>0.742±0.004</b>	<b>0.783±0.017</b>

To quantitatively analyze the result, we further compare our method with three clustering methods, Lloyd’s algorithm [40], Louvain method [41], and GRACE [30], in terms of purity, normalized mutual information (NMI) and homogeneity, as shown in Table 3. Specifically, for a cluster assignment  $C$  and a ground truth definition  $T$  partitioning  $N$  nodes, purity  $p_{\text{purity}}$  is defined as the percentage of the majority truth cluster within  $C$ :  $p_{\text{purity}} = \sum_{c \in C} \frac{1}{N} \max_{t \in T} |c \cap t|$ . The NMI score  $p_{\text{nmi}}$  and homogeneity score  $p_{\text{homo}}$  are given as  $p_{\text{nmi}} = \frac{H(C)}{H(C|T) - E(H(C) - H(C|T))}$  and  $p_{\text{homo}} = 1 - \frac{H(C|T)}{H(C)}$  respectively, where  $H$  denotes the entropy and  $E$  denotes expected value. We can observe that the proposed THC produces clusters with the highest quality among both deep and shallow baselines. The purity of THC averaged over 5 k-fold runs is 88.9%, which is 4.7% higher than the highest baseline. Similarly, the homogeneity score of THC surpasses the highest baseline by a margin of 7.2%. These results further support that the proposed clustering THC can capture the complex structure of brain networks.

## 6. CONCLUSION

In this work, we present THC, an interpretable transformer model for brain network analysis. Hierarchical clustering layers are designed based on the attention mechanism of the transformer encoder to learn a global clustering assignment in an end-to-end manner. A future direction of our work is to investigate the effect of overlapping communities on the clustering result as well as the lesion predictions.

## 7. COMPLIANCE WITH ETHICAL STANDARDS

The ABCD and ABIDE datasets employed in this study are owned by a third-party organization, where informed consent was obtained for all subjects. The data processed is anonymous with no personally identifiable information. All studies are conducted according to the Good Clinical Practice guidelines and U.S. 21 CFR Part 50 (Protection of Human Subjects), and under the approval of Institutional Review Boards.

## References

- [1] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, and J. Leskovec, "Open graph benchmark: Datasets for machine learning on graphs," *NeurIPS*, 2020.
- [2] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *ICLR*, 2019.
- [3] H. Cui, W. Dai, Y. Zhu, X. Kan, A. A. Chen Gu, J. Lukemire, L. Zhan, L. He, Y. Guo, and C. Yang, "BrainGB: A Benchmark for Brain Network Analysis with Graph Neural Networks," *IEEE TMI*, 2022.
- [4] H. Jiang, P. Cao, M. Xu, J. Yang, and O. Zaiane, "Hi-gcn: a hierarchical graph convolution network for graph embedding learning of brain network and brain disorders prediction," *Computers in Biology and Medicine*, vol. 127, p. 104096, 2020.
- [5] X. Kan, Y. Kong, T. Yu, and Y. Guo, "Bracenet: Graph-embedded neural network for brain network analysis," in *IEEE Big Data*, 2022.
- [6] Y. Yu, X. Kan, H. Cui *et al.*, "Learning task-aware effective brain connectivity for fmri analysis with graph neural networks," *arXiv preprint arXiv:2211.00261*, 2022.
- [7] Y. Yang, Y. Zhu *et al.*, "Data-efficient brain connectome analysis via multi-task meta-learning," in *SIGKDD*, 2022.
- [8] Y. Liu, L. He, and B. e. a. Cao, "Multi-view multi-graph embedding for brain network clustering analysis," *AAAI*, 2018.
- [9] X. Li, Y. Zhou, and N. D. *et al.*, "Braingnn: Interpretable brain graph neural network for fmri analysis," *Med Image Anal*, vol. 74, p. 102233, 2021.
- [10] W. L. Hamilton, "Graph representation learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 14, pp. 1–159, 2020.
- [11] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *NAACL*, 2019.
- [12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2021.
- [13] D. Kreuzer, D. Beaini, W. L. Hamilton *et al.*, "Rethinking graph transformers with spectral attention," in *NeurIPS*, 2021.
- [14] X. Kan, W. Dai, H. Cui, Z. Zhang, Y. Guo, and C. Yang, "Brain network transformer," in *NeurIPS*, 2022.
- [15] T. J. Akiki and C. G. Abdallah, "Determining the hierarchical architecture of the human brain using subject-level clustering of functional networks," *Scientific Reports*, vol. 9, pp. 1–15, 2019.
- [16] T. D. Satterthwaite, D. H. Wolf, D. R. Roalf, K. Ruparel, G. Erus, S. Vandekar, E. D. Gennatas, M. A. Elliott, A. Smith, H. Hakonarson, R. Verma, C. Davatzikos, R. E. Gur, and R. C. Gur, "Linked Sex Differences in Cognition and Functional Connectivity in Youth," *Cerebral Cortex*, vol. 25, pp. 2383–2394, 2015.
- [17] D. Meunier, R. Lambiotte, A. Fornito, K. Ersche, and E. T. Bullmore, "Hierarchical modularity in human brain functional networks," *Frontiers in neuroinformatics*, vol. 3, p. 37, 2009.
- [18] A. Alexander-Bloch, R. Lambiotte, B. Roberts, J. Giedd, N. Gogtay, and E. Bullmore, "The discovery of population differences in network community structure: new methods and applications to brain functional networks in schizophrenia," *Neuroimage*, vol. 59, 2012.
- [19] O. Sporns and R. F. Betzel, "Modular brain networks," *Annual review of psychology*, vol. 67, p. 613, 2016.
- [20] L. Nanetti, L. Cerliani, V. Gazzola, R. Renken, and C. Keysers, "Group analyses of connectivity-based cortical parcellation using repeated k-means clustering," *Neuroimage*, vol. 47, pp. 1666–1677, 2009.
- [21] V. A. Traag, L. Waltman, and N. J. Van Eck, "From louvain to leiden: guaranteeing well-connected communities," *Nature*, vol. 9, 2019.
- [22] T. J. Akiki and C. G. Abdallah, "Determining the hierarchical architecture of the human brain using subject-level clustering of functional networks," *Nature*, vol. 9, p. 19290, 2019.
- [23] X. Kan, H. Cui, L. Joshua, Y. Guo, and C. Yang, "Fbnetgen: Task-aware gnn-based fmri analysis via functional brain network generation," in *MIDL*, 2022.
- [24] H. Cui, W. Dai, Y. Zhu, X. Li, L. He, and C. Yang, "Interpretable graph neural networks for connectome-based brain disorder analysis," in *MICCAI*, 2022.
- [25] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu, "Do transformers really perform badly for graph representation?" in *NeurIPS*, 2021.
- [26] Y. Zhu, H. Cui, L. He, L. Sun, and C. Yang, "Joint embedding of structural and functional brain networks with graph neural networks for mental illness diagnosis," in *EMBC*, 2022.
- [27] G. Luo, C. Li, and Others, "Multi-view brain network analysis with cross-view missing network generation," in *BIBM*, 2022.
- [28] J. X. O'reilly, C. F. Beckmann, V. Tomassini, N. Ramnani, and H. Johansen-Berg, "Distinct and overlapping functional zones in the cerebellum defined by resting state functional connectivity," *Cerebral cortex*, vol. 20, pp. 953–965, 2010.
- [29] R. Ying, J. You, C. Morris, X. Ren, W. L. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," *NeurIPS*, 2018.
- [30] C. Yang, L. Liu, M. Liu, Z. Wang, C. Zhang, and J. Han, "Graph clustering with embedding propagation," in *IEEE Big Data*, 2020.
- [31] A. Duval and F. D. Malliaros, "Higher-order clustering and pooling for graph neural networks," in *CIKM*, 2022.
- [32] D. Chen, Y. Lin, W. Li, P. Li, J. Zhou, and X. Sun, "Measuring and relieving the over-smoothing problem for graph neural networks from the topological view," in *AAAI*, 2020.
- [33] A. Vaswani, N. Shazeer, and N. e. a. Parmar, "Attention is all you need," *NeurIPS*, 2017.
- [34] B. Casey, T. Cannonier, and M. I. C. *et al.*, "The adolescent brain cognitive development (abcd) study: Imaging acquisition across 21 sites," *Developmental Cognitive Neuroscience*, vol. 32, pp. 43–54, 2018.
- [35] M. F. Glasser and S. N. e. a. Sotiropoulos, "The minimal preprocessing pipelines for the human connectome project," *Neuroimage*, vol. 80, pp. 105–124, 2013.
- [36] C. Cameron and B. Y. *et al.*, "The neuro bureau preprocessing initiative: open sharing of preprocessed neuroimaging data and derivatives," *Frontiers in Neuroinformatics*, 2013.
- [37] R. C. Craddock, G. A. James, P. E. Holtzheimer III, X. P. Hu, and H. S. Mayberg, "A whole brain fmri atlas generated via spatially constrained spectral clustering," *Human Brain Mapping*, pp. 1914–1928, 2012.
- [38] J. Kawahara and C. J. e. a. Brown, "Brainnetcn: Convolutional neural networks for brain networks; towards predicting neurodevelopment," *NeuroImage*, vol. 146, pp. 1038–1049, 2017.
- [39] A. Doll, B. K. Hölzel, C. C. Boucard, A. M. Wohlschläger, and C. Sorg, "Mindfulness is associated with intrinsic functional connectivity between default mode and salience networks," *Frontiers in human neuroscience*, vol. 9, p. 461, 2015.
- [40] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.
- [41] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics: theory and experiment*, vol. 2008, p. P10008, 2008.