

---

# Do We Really Need Message Passing in Brain Network Modeling?

---

Liang Yang<sup>1</sup> Yuwei Liu<sup>1</sup> Jiaming Zhuo<sup>1</sup> Di Jin<sup>2</sup> Chuan Wang<sup>3</sup> Zhen Wang<sup>4</sup> Xiaochun Cao<sup>5</sup>

## Abstract

Brain network analysis plays a critical role in brain disease prediction and diagnosis. Graph mining tools have made remarkable progress. Graph neural networks (GNNs) and Transformers, which rely on the message-passing scheme, recently dominated this field due to their powerful expressive ability on graph data. Unfortunately, by considering brain network construction using pairwise Pearson’s coefficients between any pairs of ROIs, model analysis and experimental verification reveal that *the message-passing under both GNNs and Transformers can NOT be fully explored and exploited*. Surprisingly, this paper observes the significant performance and efficiency enhancements of the Hadamard product compared to the matrix product, which is the matrix form of message passing, in processing the brain network. Inspired by this finding, a novel Brain Quadratic Network (BQN) is proposed by incorporating quadratic networks, which possess better universal approximation properties. Moreover, theoretical analysis demonstrates that BQN implicitly performs community detection along with representation learning. Extensive evaluations verify the superiority of the proposed BQN compared to the message-passing-based brain network modeling. Source code is available at <https://github.com/LYJUN/BQN-demo>.

## 1. Introductions

Brain network analysis provides a deep understanding of human brain organizations and assists in neurological disease diagnosis (Fornito et al., 2016; Cui et al., 2022). As a general-purpose language to model complex relationships, the research on graphs has a long history, ranging from graph theory (Bondy & Murty, 2008), to network science (Barabási, 2013), to network embedding (Cui et al., 2019). Along with the concept of the brain network, graph mining is widely used in brain network analysis, including concepts, characteristics, and models (Fornito et al., 2016; Chung, 2019). Network motif (Sporns & Kötter, 2004), small-world property (Bassett & Bullmore, 2006), and modularity (Meunier et al., 2009) are identified from the perspective of characteristics, while the persistent homology (Bendich et al., 2016), community detection (Garcia et al., 2018) and null model (Váša & Mišić, 2022) are employed to analyze brain network from the perspective of the model.

Graph Learning (Xia et al., 2021) plays an important role in diverse machine learning fields. (1) For tasks on graph data, Graph Neural Networks (GNNs), which combine graph topology and node attribute for representation, recently dominated this field (Wu et al., 2021b). They follow the message-passing scheme (Gilmer et al., 2017) by propagating node representation as a message over the graph. (2) For the domains without explicit graph structure, such as natural language processing (NLP) and computer vision (CV), implicit graph learning also shows amazing performance. The key component Multi-Heads Self-Attention (MHSA) in Transformer (Vaswani et al., 2017) and its variant ViT (Dosovitskiy et al., 2021) constructs a fully connected graph between all token pairs and performs a holistic aggregation (Section 3.3). In conclusion, the message-passing scheme dominates many machine learning fields.

Interactions between brain regions are regarded as the key factors for neural development and disorder analysis. Functional MRI (fMRI) provides valuable information for exploring connectivity by capturing correlations between signal sequences of brain regions. Thus, it is direct to analyze brain networks, which are often constructed using pairwise Pearson correlation coefficients between any pairs of ROIs, using powerful graph learning tools, such as GNNs (Li et al., 2021; Cui et al., 2022; Bessadok et al., 2023) and Trans-

<sup>1</sup>Hebei Province Key Laboratory of Big Data Calculation, School of Artificial Intelligence, Hebei University of Technology, Tianjin, China <sup>2</sup>Tianjin Key Laboratory of Cognitive Computing and Application, College of Intelligence and Computing, Tianjin University, Tianjin, China <sup>3</sup>School of Computer Science and Technology, Beijing JiaoTong University, Beijing, China <sup>4</sup>School of Artificial Intelligence, OPtics and ElectroNics (OPEN), School of Cybersecurity, Northwestern Polytechnical University, Xi'an, China <sup>5</sup>School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University, Shenzhen, China. Correspondence to: Jiaming Zhuo <jiaming.zhuo@outlook.com>.

*Proceedings of the 42<sup>nd</sup> International Conference on Machine Learning*, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

formers (Yu et al., 2024). Unfortunately, following model analysis and experiments motivate questioning this direct treatment in the brain network.

- GNN-based methods constructed node attributes from the brain network topology, which goes against GNNs' need for different types of information, and thus can't be fully exploited GNNs' characteristics (Fig. 1).
- Transformers employ the constructed brain network containing a holistic relationship between all ROIs, as initial token embedding. Thus the necessity to explore holistic relations with Transformers is weak (Fig. 1).
- Simple classifier with **brain functional connectivity matrix** as feature outperforms basic GNNs and Transformers (Fig. 2).

To enhance the performance of brain network analysis, the fundamental brain network operator should be updated. The recent breakthrough points out that the quadratic function can implement XOR logic operation and possesses better universal approximation properties compared to the linear function (Fan et al., 2018; 2020). Inspired by this, the quadratic network is employed in brain network analysis to obtain the Brain Quadratic Network (BQN). BQN iteratively performs Hadamard product/element-wise product between the representation in the previous layer and the initial representation to get a new representation. Theoretical analysis reveals that BQN is equivalent to the updating rule of a community detection objective function based on nonnegative matrix factorization (NMF) of the adjacency matrix. Since the NMF of the adjacency matrix is a widely-used community detection paradigm, the proposed BQN seeks representation, which can capture mesoscopic community structure to reflect brain functional modules. The main contributions of this paper are summarized as follows.

- We investigate the rationality of the widely adopted message-passing in the brain network analysis.
- We propose a simple and effective Brain Quadratic Network (BQN) as the fundamental operator, **with superior computational efficiency**.
- We provide rigorous theoretical analysis connecting the proposed BQN with community detection.
- The proposed BQN achieves new SOTA on widely-used brain datasets.

## 2. Related Work

Recent advancements in brain network analysis have led to a surge in the application of graph-based learning methods, especially Graph Neural Networks and Graph Transformers.

**Graph Neural Networks.** In recent studies, Graph Neural Networks (GNNs) have been leveraged to learn representations of brain regions, as well as the intricate patterns

of functional connectivity within the brain. For example, BrainGNN (Li et al., 2021) utilizes ROI-aware GNNs and specialized pooling operators to identify critical nodes, improving the model's interpretability. BrainGB (Cui et al., 2022) systematically assesses various GNN designs on brain network data and introduces a practical pipeline that enhances GNN applications in brain research. FBNetGen (Kan et al., 2022a) constructs a learned task-oriented brain graph for downstream tasks. A-GCL (Zhang et al., 2023) proposes an adversarial self-supervised brain graph neural network by integrating graph contrastive learning with ROIs multiband information. However, these approaches often concentrate on local neighborhood aggregation, potentially neglecting the significance of interactions between non-adjacent brain regions.

**Graph Transformers.** Graph Transformers are utilized in the analysis of brain networks, with the objective of capturing the hostile interaction of brain regions. BrainNETTF (Kan et al., 2022b) employs classical Transformer encoders to generate ROI embeddings based on Pearson correlation matrices, along with an orthogonal clustered readout. ALTER (Yu et al., 2024) has specifically designed a brain graph Transformer to capture long-range dependencies among brain regions. ContrastPool (Xu et al., 2024) emphasizes attention on ROIs and subjects to learn a contrast graph, guiding the generation of brain representations through graph pooling. BioBGT (Peng et al., 2025) captures the small-world architecture in brain graphs through a network entanglement-based technique, emphasizing the biological characteristics of brain structure.

However, many of these methods often conflate the concepts of brain functional connectivity matrices and region of interest (ROI) features. The correlation coefficient matrix serves a dual purpose: it acts as both the adjacency matrix of the brain graph and the feature matrix. This duality contradicts established principles in graph research, where topology and features are typically regarded as two distinct types of information. This observation prompts us to design a simple yet effective network architecture specifically tailored for brain functional connectivity.

## 3. Preliminaries

This section provides notations, problem definition, and the concepts of GNNs and Transformers.

### 3.1. Notations and Problem Definition

Brain network, which models the connectivity between ROIs, can be represented as a graph  $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ , where  $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$  stands for the collection of  $N$  nodes, i.e., ROIs,  $\mathcal{E}$  denotes the collection of edges between nodes, and  $\mathbf{X} \in \mathbb{R}^{N \times D}$  represents the feature matrix with the

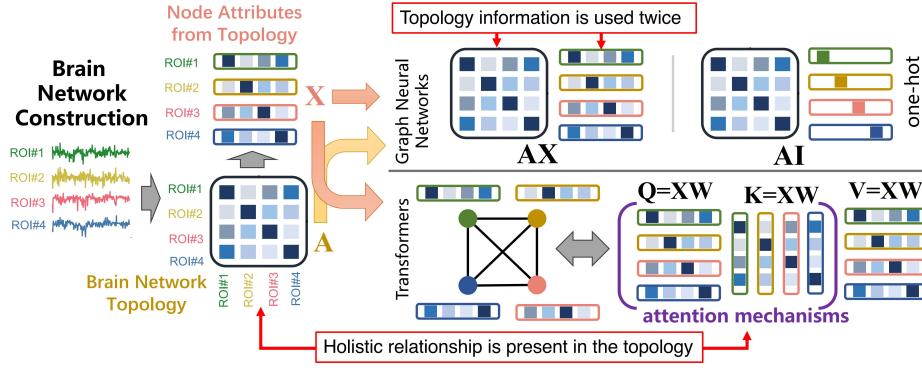


Figure 1. Brain network construction and the GNNs and Transformers on brain network analysis. In brain network construction, node attributes are from constructed brain network topology. Thus, GNNs utilize this information twice. The brain network already contains the correlation between all ROIs, while the attention mechanism also tends to learn this.

$i$ -th row  $\mathbf{x}_i \in \mathbb{R}^D$  as the feature of node/ROI  $v_i$ . The graph topology is represented as the adjacency matrix  $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$ , where  $a_{i,j}$  is the weight between nodes  $v_i$  and  $v_j$ .  $\mathcal{N}(v)$  denotes the neighbourhoods of node  $v$ .  $\mathbf{D} \in \mathbb{R}^{N \times N}$  stands for the degree matrix with diagonal element  $d_{vv} = \sum_{u \in \mathcal{N}(v)} a_{uv}$  as the degree of node  $v$ .

For brain network analysis tasks, a set of  $L$  subjects' brain network  $\mathcal{G} = \{G_1 \dots G_L\}$  and the corresponding labels  $\mathcal{Y} = \{y_1 \dots y_L\}$ , which indicates the presence of a disease, biological sex or other properties of the subject, are provided. Brain network modeling aims to learn from given data  $\mathcal{G}$  and  $\mathcal{Y}$  by designing a function  $y = f(h(G))$ , which is composed of a representation learning function  $h_G = h(G)$  and a prediction function  $y = f(h_G)$  based on the learned brain representation  $h_G$ .

### 3.2. Graph Neural Networks

Graph Neural Networks (GNNs) often follow the message-passing paradigm (Gilmer et al., 2017; Yang et al., 2022; Zhuo et al., 2023), by iteratively aggregating node representations from neighborhoods and combining them with the representation of itself. By denoting  $\mathbf{H}^l$  as the collection of representations in the  $l$ -th layer, its  $v$ -th row  $\mathbf{h}_v^l$ , i.e., the representation of node  $v$  in the  $l$ -th layer, can be obtained as

$$\hat{\mathbf{h}}_v^l := \text{Aggregation}^l(\{\mathbf{h}_u^{l-1} | u \in \mathcal{N}(v)\}), \quad (1)$$

$$\mathbf{h}_v^l := \text{Combination}^l(\mathbf{h}_v^{l-1}, \hat{\mathbf{h}}_v^l), \quad (2)$$

where  $\text{Aggregation}(\cdot)$  and  $\text{Combination}(\cdot)$  denote the aggregation and combination modules, respectively. Following the above paradigm, classical GNNs, e.g., GCN (Kipf & Welling, 2017)-one of representative GNNs employ the weighted average function  $\tilde{\mathbf{A}}\mathbf{H}^{l-1}$  to implement the above operations as

$$GCN : \quad \mathbf{H}^l = \sigma(\tilde{\mathbf{A}}\mathbf{H}^{l-1}\mathbf{W}^l), \quad (3)$$

where  $\sigma$  denotes the non-linear activation function and  $\mathbf{H}^0 = \mathbf{X}$  denotes the initial node attributes.  $\tilde{\mathbf{A}} = (\mathbf{D} + \mathbf{I}_n)^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I}_n)(\mathbf{D} + \mathbf{I}_n)^{-\frac{1}{2}}$  represents the normalized adjacency matrix with self-loop.

### 3.3. Transformers

Unlike GNNs that aggregate information from local neighborhoods, Transformers facilitate a holistic aggregation across all token pairs through the self-attention mechanism. The main component of Transformer (Vaswani et al., 2017) is Multi-Heads Self-Attention (MHSA). Given  $N$  tokens input  $\mathbf{Z} = [\mathbf{z}_i]_{i=0}^{N-1} \in \mathbb{R}^{N \times F}$ , which is the concatenation of the initial token embedding  $\mathbf{X}$  and the position encoding  $\mathbf{P}$  as  $\mathbf{Z} = [\mathbf{X}||\mathbf{P}]$ , the self-attention first maps the input features  $\mathbf{Z}$  to query ( $\mathbf{Q}$ ), key ( $\mathbf{K}$ ), and value ( $\mathbf{V}$ ) vectors. Then, attention scores from query-key pairs are employed to aggregate the value vectors in a weighted manner, operating as a global message-passing mechanism. Specifically, a Self-Attention (SA) module can be formulated as

$$\begin{aligned} \mathbf{Q} &= \mathbf{Z}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{Z}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{Z}\mathbf{W}_V, \\ \hat{\mathbf{Z}} &= \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{F}}\right)\mathbf{V}, \end{aligned} \quad (4)$$

where  $\mathbf{W}_Q$ ,  $\mathbf{W}_K$ , and  $\mathbf{W}_V \in \mathbb{R}^{F \times F}$  denotes trainable projection matrices and  $F$  stands for the feature dimensions. Further details of GNNs and Transformers are provided in the appendix A.

**Graph Transformer.** To employ a Transformer to graph structure data, it is critical to incorporate graph topology appropriately (Zhuo et al., 2025). There are two kinds of strategies (Ying et al., 2021a; Rampásek et al., 2022). Some methods directly employ adjacency matrix  $\mathbf{A}$  to regularize the Transformer, e.g., element-wise product  $\mathbf{A} \odot \text{Sim}^{QK}$ . The other methods encode topology structure into position encoding  $\mathbf{P}$ , e.g., the eigenvector of its Laplacian matrix. Thus, it is evident that the utilization of the Transformer

for graph-data processing calls for a careful handling of graph topology and node representation, a consideration that stands out as especially striking in the context of brain network analysis.

## 4. Issues with Message Passing

This section begins with the method of brain network construction, followed by the analysis of the GNNs and Transformers on the constructed brain network. Finally, experiments are conducted to verify the above analysis.

### 4.1. Brain Network Construction

Refer to (Cui et al., 2022) for the detailed construction process of the brain network from raw data. Here, the final two steps, which are closely related to the brain network  $G$ , are considered. Firstly, the Brain Region Parcellation segments each subject into ROIs. Secondly, the weights of edges between ROIs are calculated using pairwise Pearson correlation coefficient as

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}, \quad (5)$$

where  $\{x_1, \dots, x_n\}$  and  $\{y_1, \dots, y_n\}$  are two sequences of response values of two ROIs with the same length  $n$ , and  $\bar{x}$  and  $\bar{y}$  are the means of sequences. To obtain a robust graph, a threshold is often used to sparsify the edge weights as

$$a_{xy} = \begin{cases} r_{xy}, & \text{if } r_{xy} > \text{threshold}, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

The constructed symmetric  $\mathbf{A} = [a_{xy}] \in \mathbb{R}^{N \times N}$  is seen as the adjacency matrix of the brain network. Note that the sequence  $\{x_1, \dots, x_n\}$  is infrequently characterized as the attribute of ROI, since its content is the response values at some moments instead of the essential characteristics of the ROI. Section 4.2 elaborates on how GNNs and Transformers construct node attributes.

### 4.2. Model Analysis

This section analyzes how GNNs and Transformers process brain networks constructed in Section 4.1.

**GNNs for Brain Network.** As shown in Section 3.2, GNNs learn representation by combining graph topology and node attributes, which are two different types of information. Unfortunately, the constructed brain network often lacks essential node attributes. To alleviate this issue, existing GNN-based methods often constructed node attributes from the brain network topology, such as (1) identity matrix, (2) Eigenvectors of the adjacency matrix, (3) node degree, (4) local statistic of degree, (5) adjacency matrix itself or from another correlation measurement (Fig. 1). This strategy

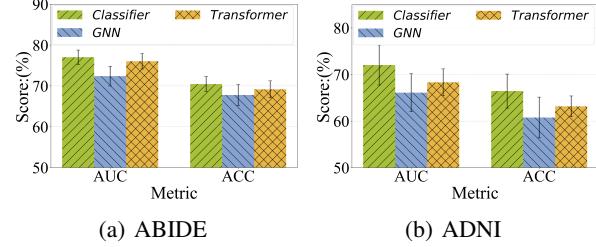


Figure 2. Performances of basic classifier, GNN and Transformer on ABIDE (left) and ADNI (right) datasets according to AUC and ACC. Note that the simple classifier consistently outperforms basic GNN and Transformer with message-passing strategy.

goes against GNNs' need for different types of information. Therefore, GNNs' characteristics, especially message passing, can't be fully exploited in brain network modeling.

**Transformers for Brain Network.** As shown in Section 3.3, Transformers learn holistic relations between all tokens with the self-attention mechanism. To employ Transformers in the brain network modeling, the initial token embedding is often set as the adjacency matrix, i.e.,  $\mathbf{X} = \mathbf{A}$ . Since  $\mathbf{A}$  is obtained by calculating Pearson correlation coefficients between all ROI pairs, the holistic relationship between all tokens is present in the initial token embedding  $\mathbf{X}$ . Thus, the necessity of employing Transformers, especially the implicit message passing in the attention mechanism, to explore holistic relations in brain networks is weak (Fig. 1).

In summary, the message-passing under both GNNs and Transformers can't be fully explored and exploited by considering the characteristics of the brain networks.

### 4.3. Experimental Investigation

To verify the above analysis of the ability of GNNs and Transformers in brain networks, this section conducts basic experiments. The performances are measured according to AUC and ACC metrics on ABIDE and ADNI datasets, results of more metrics are placed in Appendix B. Three baselines are involved as follows:

- The simple classifier  $f(\mathbf{A}) = \mathbf{AW}$ , with  $\mathbf{A}$  as the feature matrix and  $\mathbf{W}$  as learnable matrix;
- The GNN as in Section 3.2 with  $\mathbf{A}$  as topology and node attribute;
- The Transformer as in Section 3.3 with  $\mathbf{A}$  as node attribute.

The performances are shown in Fig. 2. It can be observed that the basic and simple classifier consistently outperforms the other two models, i.e., GNN and Transformer. This result meets our analysis in Section 4.2 that message passing under both GNNs and Transformers can't be fully explored and exploited in brain network modeling. Note that some GNN-based and Transformer-based models, such as Brain

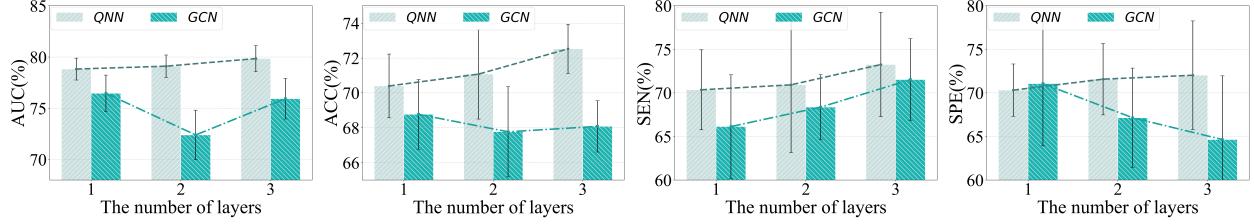


Figure 3. Performance comparison of QNN (Hadamard product) and GCN (matrix product) on ABIDE dataset according to AUC, ACC, SEN, and SPE. Note that QNN consistently outperforms GCN, both the AUC and the ACC increase as the number of layers increases.

Network Transformer (Kan et al., 2022b), achieve much better performance compared to the basic GNN and Transformer. However, their high performances mainly rely on the modification to GNN and Transformer instead of the message-passing mechanism behind them.

## 5. Methodology

Brain Quadratic Network (BQN), which breaks the message-passing scheme, is proposed here.

### 5.1. Motivations

The model analysis and experimental investigation in Sections 4.2 and 4.3 demonstrate that message passing under both GNNs and Transformers can't be fully explored and exploited in brain network modeling. Thus, to enhance the performance of brain network analysis, the fundamental brain network operator should be updated.

Here, the simple classifier  $f(\mathbf{A}) = \mathbf{AW}$  is employed as the base of the investigation for its satisfactory performance. For one-dimension output,  $f(\cdot)$  acts as the linear function as  $f(\mathbf{a}) = \sum_{i=1}^N w_i a_i = \mathbf{wa}$ , where  $\mathbf{a} = [a_1, \dots, a_N]$  and  $\mathbf{w} = [w_1, \dots, w_N]$  denote one-row of  $\mathbf{A}$  and one-column of  $\mathbf{W}$ , respectively. It is well-known that linear functions can't implement XOR logic operation (Bishop & Nasrabadi, 2006). Recent attempts (Fan et al., 2018; 2020) demonstrate the expressive ability of the Quadratic/Second-order function of the following forms

$$\begin{aligned} h(\mathbf{a}) &= \left( \sum_{i=1}^N w_{ri} a_i + b_r \right) \left( \sum_{i=1}^N w_{gi} a_i + b_g \right) + \sum_{i=1}^N w_{oi} a_i^2 + c \\ &= (\mathbf{w}_r \mathbf{a} + b_r)(\mathbf{w}_g \mathbf{a} + b_g) + \mathbf{w}_o(\mathbf{a}^2) + c, \end{aligned} \quad (7)$$

where  $\mathbf{w}_r = [w_{r1}, \dots, w_{rN}]$ ,  $\mathbf{w}_g = [w_{g1}, \dots, w_{gN}]$ ,  $\mathbf{w}_o = [w_{o1}, \dots, w_{oN}]$ ,  $b_r$ ,  $b_g$ , and  $c$  are learnable parameters.  $\mathbf{a}^2 = \mathbf{a} \odot \mathbf{a}$  denotes the element-wise square operator, and  $\odot$  stands for Hadamard product product. It is proved that quadratic function can implement XOR logic operation and possesses some better universal approximation properties compared to linear function (Fan et al., 2020; 2025). This motivates us to employ the quadratic function to enhance the performance of brain network analysis. Moreover, we

provide further elaboration of Quadratic function and its representative applications in Appendix C.

To verify the effectiveness of the quadratic function with the Hadamard product, the following two brain network encoding methods are compared with  $\mathbf{H}_0 = \mathbf{AW}^0$  as the initial embedding (0-th layer).

- Graph Convolution Network:  $\mathbf{H}^l = \sigma(\mathbf{AH}^{l-1}\mathbf{W}^l)$ ,
- Quadratic Neural Network:  $\mathbf{H}^l = \mathbf{H}^{l-1} \odot (\mathbf{AW}^l)$ ,

where  $\odot$  stands for Hadamard product/element-wise product. The performances of the Graph Convolution Network (GCN) and the Quadratic Neural Network (QNN) are shown in Fig. 3. It can be observed that the QNN significantly and consistently outperforms the GCN. Besides, both the AUC and the ACC increase as the number of layers increases.

In summary, both the expressive ability and universal approximation from theory and experimental results suggest the Quadratic Neural Network (QNN) as an alternative for brain network modeling.

### 5.2. Brain Quadratic Network

According to Eq. (7), the Quadratic Network for Brain is formulated by using the Hadamard product as follows.

$$\begin{aligned} \mathbf{H}^l &= (\mathbf{H}^{l-1}\mathbf{W}_A^l) \odot (\mathbf{H}^{l-1}\mathbf{W}_B^l) \\ &\quad + (\mathbf{H}^{l-1} \odot \mathbf{H}^{l-1}) \mathbf{W}_C^l, \end{aligned} \quad (8)$$

where  $\mathbf{W}_A^l$  and  $\mathbf{W}_B^l$  are the learnable parameters for the  $l$ -th layer. To stabilize the learning process, the first term of Eq. (8) is replaced with  $\mathbf{H}^{l-1} \odot (\mathbf{AW}_A^l)$ , which meets the QNN formula in Section 5.1, and the Brain Quadratic Network(BQN) is as follows:

$$\mathbf{H}^l = \mathbf{H}^{l-1} \odot (\mathbf{AW}_A^l) + (\mathbf{H}^{l-1} \odot \mathbf{H}^{l-1}) \mathbf{W}_H^l, \quad (9)$$

where learnable mappings  $\mathbf{W}_A^l$  and  $\mathbf{W}_H^l$  are for original brain network  $\mathbf{A}$  and the embedding  $\mathbf{H}$ . As shown in the experiments in Section 5.1, the first term in Eq. (9) plays a key role in performance enhancement, while the second one mainly tends to reduce the variance.

### 5.3. Theoretical Analysis

This section provides deep insight into the Brain Quadratic Network, especially the first term in Eq. (9), i.e.,

$$\mathbf{H}^l = \mathbf{H}^{l-1} \odot (\mathbf{A}\mathbf{W}_A^l). \quad (10)$$

The following theorem provides an intuitive understanding from the concept of community detection.

**Theorem 5.1.** *The iterative formula Eq. (10) of the proposed BQN is equivalent to the updating rule of the following community detection objective function, which is based on nonnegative matrix factorization (NMF) of the adjacency matrix.*

$$\min_{\mathbf{H} \geq 0, \mathbf{W}_A \geq 0} \mathcal{L}(\mathbf{H}) = \|\mathbf{H}\mathbf{W}_A^\top - \mathbf{A}\|_F^2, \quad (11)$$

where  $\mathbf{H} \geq 0$  denotes all elements in  $\mathbf{H}$  are non-negative.

*Proof.* The constrained optimization in Eq. (11) can be converted to an unconstrained one by employing a Lagrange multiplier (Bertsekas, 2014). To this end, a Lagrange multiplier matrix  $\Theta$  is introduced corresponding to the nonnegative constraints  $\mathbf{H} \geq 0$ . Thus, the equivalent objective function is

$$\min_{\mathbf{H}} \mathcal{L}(\mathbf{H}) = \|\mathbf{H}\mathbf{W}_A^\top - \mathbf{A}\|_F^2 + \text{tr}(\Theta\mathbf{H}^\top).$$

Set derivative of  $\mathcal{L}(\mathbf{H})$  with respect to  $\mathbf{H}$  to 0, it obtains

$$\Theta = 2\mathbf{A}\mathbf{W}_A - 2\mathbf{H}\mathbf{W}_A^\top\mathbf{W}_A. \quad (12)$$

Following the KKT condition for the non-negativity of  $\mathbf{H}$ , Eq. (12) can be further reformulated as element-wise form

$$(2\mathbf{A}\mathbf{W}_A - 2\mathbf{H}\mathbf{W}_A^\top\mathbf{W}_A)_{ij} \mathbf{H}_{ij} = \Theta_{ij} \mathbf{H}_{ij} = 0, \quad (13)$$

which is the fixed point equation that the solution must satisfy at convergence. Given an initial value of  $\mathbf{H}$ , the updating rule of  $\mathbf{H}$  can be formulated as

$$\mathbf{H}_{ij} \leftarrow \mathbf{H}_{ij} \frac{(\mathbf{A}\mathbf{W}_A)_{ij}}{(\mathbf{H}\mathbf{W}_A^\top\mathbf{W}_A)_{ij}}, \quad (14)$$

where the denominator is to normalize the iterations, and  $\leftarrow$  denotes the assignment operation. Therefore, the iteration can be written as in matrix form

$$\mathbf{H} = \mathbf{H} \odot (\mathbf{A}\mathbf{W}_A), \quad (15)$$

which is the same as the iterative formula of BQN.  $\square$

The term  $\|\mathbf{H}\mathbf{W}_A^\top - \mathbf{A}\|_F^2$  in Eq. (11) is the nonnegative matrix factorization of the adjacency matrix  $\mathbf{A}$ . Nonnegative matrix factorization of the adjacency matrix is a widely-used community detection method (Yang & Leskovec, 2013;

Wang et al., 2016; Luo et al., 2022), since its superior performance and outstanding interpretability. Therefore, the formula Eq. (10) leads the proposed BQN to seek representation, which can capture mesoscopic community structure to reflect brain functional modules. This provides an interpretation of the success of the proposed BQN. In addition to the theoretical analysis, experiments reflecting the clustering characteristics of brain region representations obtained through stacking multiple layers of Eq. (9) are presented in Appendix D.

## 6. Evaluations

### 6.1. Experiments Setup

**Datasets.** In the experiments, two real-world fMRI datasets are employed: as follows.

- *Autism Brain Imaging Data Exchange (ABIDE):* This dataset is primarily utilized to investigate brain functional connectivity variation and structural differences associated with Autism Spectrum Disorder. The pre-processed data version can be accessed from the official website<sup>1</sup>. It collects resting-state functional magnetic resonance imaging data from 17 international sites, as well as anatomical and phenotypic data. The dataset we used contains 516 Autism Spectrum Disorder patients (ASD) and 493 normal controls (NC).
- *Alzheimer's Disease Neuroimaging Initiative (ADNI):* ADNI is a widely utilized multimodal neuroimaging repository focused on Alzheimer's disease. The raw images can be obtained from ADNI official website<sup>2</sup>. Access is limited and requires adherence to a request procedure to acquire the data. The dataset for this paper contains 53 Alzheimer's disease (AD) samples and 71 normal controls.

The construction of brain functional connectivity matrix for ABIDE is based on Craddock 200 atlas. The Pearson correlation coefficient between the region-averaged BOLD signals from pairs of ROIs (Regions of Interest) is adopted as the measure of functional connectivity strength between ROIs, namely as brain graph adjacency matrix (Cui et al., 2022; Bessadok et al., 2023; Xu et al., 2024). For ADNI, the fMRI data are first preprocessed using the Data Processing Assistant for Resting-State fMRI (DPARSF) toolkit. Next brain ROIs are defined based on AAL 90 atlas and the average time-series feature is calculated for each individual brain ROI. Pearson correlation coefficient between ROIs is then calculated, which serves as the functional connectivity matrix. Specifically, thresholds are set to keep edges with positive weights and drop those with negative weights.

<sup>1</sup><http://preprocessed-connectomes-project.org/abide/>

<sup>2</sup><https://adni.loni.usc.edu/>

Table 1. Performance comparison with two categories of baselines. The best model is bolded and the runner-up is underlined, respectively.

Type	Model	ABIDE				ADNI			
		AUC↑	ACC↑	SEN↑	SPE↑	AUC↑	ACC↑	SEN↑	SPE↑
Graph Neural Networks	GCN	59.59 $\pm$ 3.44	59.30 $\pm$ 3.38	56.67 $\pm$ 4.37	61.55 $\pm$ 5.29	62.45 $\pm$ 3.63	59.12 $\pm$ 4.53	54.55 $\pm$ 9.96	62.00 $\pm$ 8.55
	GAT	60.43 $\pm$ 3.88	60.10 $\pm$ 4.13	59.26 $\pm$ 5.51	62.89 $\pm$ 8.03	62.00 $\pm$ 2.88	58.75 $\pm$ 2.86	53.20 $\pm$ 7.50	64.79 $\pm$ 7.65
	BrainGNN	64.42 $\pm$ 3.57	63.09 $\pm$ 1.35	65.65 $\pm$ 2.88	60.67 $\pm$ 3.68	61.81 $\pm$ 1.58	58.72 $\pm$ 3.14	52.88 $\pm$ 9.70	62.70 $\pm$ 4.26
	BrainGB	70.32 $\pm$ 3.66	65.12 $\pm$ 3.90	67.01 $\pm$ 10.00	60.07 $\pm$ 8.53	66.44 $\pm$ 3.33	63.70 $\pm$ 4.65	60.73 $\pm$ 8.25	64.67 $\pm$ 9.07
	FBNETGEN	74.55 $\pm$ 3.77	67.09 $\pm$ 3.37	64.71 $\pm$ 9.85	69.61 $\pm$ 9.30	67.05 $\pm$ 2.16	63.26 $\pm$ 1.38	66.79 $\pm$ 6.93	61.31 $\pm$ 9.65
	A-GCL	73.86 $\pm$ 2.91	71.04 $\pm$ 2.40	71.42 $\pm$ 3.03	70.95 $\pm$ 3.19	68.15 $\pm$ 3.33	67.08 $\pm$ 4.57	65.45 $\pm$ 8.39	68.33 $\pm$ 4.36
Graph Transformer Models	SAN	71.35 $\pm$ 2.18	65.34 $\pm$ 2.91	55.41 $\pm$ 9.29	68.39 $\pm$ 7.50	66.11 $\pm$ 3.41	61.78 $\pm$ 4.22	53.94 $\pm$ 7.56	63.63 $\pm$ 8.51
	Graphomer	63.91 $\pm$ 4.05	61.88 $\pm$ 6.85	66.30 $\pm$ 9.98	55.74 $\pm$ 11.00	60.69 $\pm$ 5.26	55.75 $\pm$ 3.18	60.18 $\pm$ 11.36	47.75 $\pm$ 13.53
	GraphTrans	60.13 $\pm$ 6.73	57.83 $\pm$ 4.71	65.70 $\pm$ 10.3	49.77 $\pm$ 11.52	61.41 $\pm$ 3.65	58.60 $\pm$ 5.41	65.57 $\pm$ 6.05	54.37 $\pm$ 3.42
	BrainNETTF	77.93 $\pm$ 1.41	69.26 $\pm$ 2.26	65.92 $\pm$ 8.60	<b>73.20<math>\pm</math>6.06</b>	69.73 $\pm$ 2.61	67.85 $\pm$ 2.92	63.64 $\pm$ 6.27	70.67 $\pm$ 8.33
	ContrastPool	57.36 $\pm$ 0.87	57.44 $\pm$ 0.69	57.66 $\pm$ 6.85	57.08 $\pm$ 7.79	68.17 $\pm$ 3.28	66.21 $\pm$ 3.90	61.51 $\pm$ 7.44	<b>72.43<math>\pm</math>6.53</b>
	ALTER	77.99 $\pm$ 2.21	70.10 $\pm$ 2.26	72.84 $\pm$ 7.40	67.68 $\pm$ 5.81	<u>71.86<math>\pm</math>3.13</u>	66.92 $\pm$ 3.93	<b>71.55<math>\pm</math>8.91</b>	64.00 $\pm$ 6.80
	BioBGT	69.96 $\pm$ 1.18	69.70 $\pm$ 2.90	67.04 $\pm$ 3.41	72.02 $\pm$ 4.67	63.16 $\pm$ 3.74	62.27 $\pm$ 3.23	63.97 $\pm$ 7.88	60.55 $\pm$ 6.71
BQN (Ours)		<b>79.85<math>\pm</math>1.27</b>	<b>72.53<math>\pm</math>1.41</b>	<b>73.26<math>\pm</math>5.99</b>	72.03 $\pm$ 6.24	<u><b>74.18<math>\pm</math>3.34</b></u>	<b>68.62<math>\pm</math>3.22</b>	70.91 $\pm$ 8.60	68.00 $\pm$ 7.81

**Baselines.** Thirteen Baselines are compared in the experiments, which can be divided into two categories.

- Graph Neural Network (GNN)-based models, including two typical GNNs: GCN (Kipf & Welling, 2017) and GAT (Veličković et al., 2017)), and four brain-specific GNNs: BrainGNN (Li et al., 2021), BrainGB (Cui et al., 2022), FBNETGEN (Kan et al., 2022a) and A-GCL (Zhang et al., 2023).
- Graph Transformer (GT)-based models, including three typical GTs: SAN (Kreuzer et al., 2021), Graphomer (Ying et al., 2021b) and GraphTrans (Wu et al., 2021a)), and four brain-specific GTs: BrainNETTF (Kan et al., 2022b), ContrastPool (Xu et al., 2024), ALTER (Yu et al., 2024) and BioBGT (Peng et al., 2025).

Additional comparative experiments of BQN are conducted on supplemental datasets, please refer to Appendix E.

**Metrics.** To conduct comprehensive evaluations of performance, a combination of machine learning and medical diagnostic-specific metrics are employed, including Area Under the Receiver Operating Characteristic Curve (AUC), Accuracy (ACC), Sensitivity (SEN), and Specificity (SPE). AUC provides a threshold-independent measure of the model’s ability to distinguish between classes. ACC is a straightforward metric to assess the model’s classification performance. SEN, also known as the true positive rate, is a critical metric for medical diagnostics. In contrast, SPE represents the true negative rate.

**Implementation Details.** The experiments are performed on a GeForce RTX3090 GPU. For all datasets, we employ random splits with the ratio 7:1:2 to get the training set, validation set and test set. Furthermore, the number of training

epochs is set to 200 with batch size 16. An adam optimizer is adopted with initial learning rate of  $10^{-4}$  and weight decay as  $10^{-4}$  while training, target learning rate is from  $10^{-5}$  to  $10^{-4}$ . The activation function we selected is LeakyReLU. The number of layers, i.e.,  $k$  is selected from 1, 2, 3, 4, 5 and the dropout rate is chosen from 0., 0.1, 0.2, 0.3. The results were averaged over 5 random runs. Notably, existing studies of GNNs and Transformers for brain networks show marked differences in data processing and model selection, which poses challenges in fairly evaluating their performance. To address this, this paper unifies these settings and compares methods in a fair manner. The details of corresponding unified settings are elaborated in the Appendix F.

## 6.2. Result Analysis

**Brain Disorder Disease Classification.** The performance comparison between BQN and the baseline models on the ABIDE and ADNI datasets are presented in Tab. 1. Upon observation, it is evident that compared to GNN-based and GT-based baselines, BQN exhibits superior performance in terms of AUC and ACC metrics. In particular, on the ABIDE dataset, BQN achieves an accuracy that is 2.43% higher than that of the runner-up baseline ALTER. This advantage can be attributed to the fact that BQN directly leverages the essential information from brain functional connectivity matrices. Therefore, it reduces model complexity and inference uncertainty that may arise from extraneous factors. Moreover, during training, BQN implicitly learns characteristic representations of brain functional modules, as evidenced in Sec. 5.3. These learned representations of functional regions enhance the accuracy of brain disorder classification. By directly modeling the intrinsic properties of brain functional connectivity, BQN is particularly well-suited for brain disorder classification tasks, especially

Table 2. Comparison of running time(s) on ABIDE and ADNI.

Method	ABIDE	ADNI
GCN	14.26	4.12
FBNETGEN	19.73	5.71
A-GCL	41.57	6.42
SAN	712.01	152.23
Graphomer	973.52	179.11
BrainNETTF	17.32	4.77
ContrastPool	283.02	76.05
ALTER	37.21	7.12
BioBGT	31.24	6.78
BQN(Ours)	<b>11.31</b>	<b>3.33</b>

when dealing with the complex and limited-scale nature of neuroimaging datasets. These experimental results and analyses indicate superior performance and highlight the promising potential of BQN.

**Efficiency Test.** This experiment aims to thoroughly examine the efficiency of the proposed BQN. It achieves this by comparing the running time of BQN with those of nine representative baselines. The times reported in Tab. 2 are the total times of training 100 epochs in full batch mode. It can be observed that compared to all baselines, the proposed BQN achieves the shortest runtime on ABIDE and ADNI datasets. This is mainly attributed to the simplicity of the proposed BQN, which guarantees its efficiency. To be specific, the core computation of BQN involves element-wise multiplication of matrices via Hadamard product, which poses a quadratic complexity of  $O(n^2)$ , where  $n$  denotes ROIs number. In contrast, both GNN-based and GT-based models require the computation of matrix multiplication between two square matrices (the propagation matrix and the feature matrix, e.g., Pearson matrices), which typically involves a cubic complexity of  $O(n^3)$ .

**Additional Experiments w/o Hadamard residual.** To validate the rationality and effectiveness of residuals, i.e., the second term proposed by Eq. (9). An ablation study is conducted with an architectural theme defined as  $\mathbf{H}^{(l)} = \mathbf{H}^{(l-1)} \odot \mathbf{A}\mathbf{W}_A^l$  with or without  $(\mathbf{H}^{l-1} \odot \mathbf{H}^{l-1})\mathbf{W}_H^l$  moduel, where  $\mathbf{H}^0 = \mathbf{A}\mathbf{W}$ ,  $\mathbf{W}$  represents a linear layer. The comparison results are shown in Fig. 4. It can be seen that for the AUC, ACC, and SEN metrics, the addition of the Hadamard residual provides consistent performance improvements on both ABIDE and ADNI datasets. This illustrates the validity of the Hadamard residual. Besides, for AUC and ACC metrics, the standard deviations of the model results are smaller, indicating that the Hadamard residual also contributes to the stability of the model.

**Hyperparameter Sensitivity Analysis.** This experiment is designed to offer an intuitive understanding of how to select the optimal number of layers  $k$ . The effect of this hyperparameter on the model performance is reported in

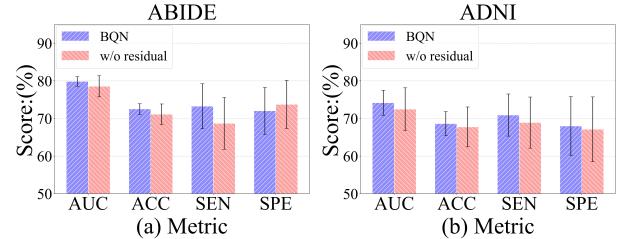


Figure 4. Performance comparison of BQN with and without Hadamard residual term on two datasets.

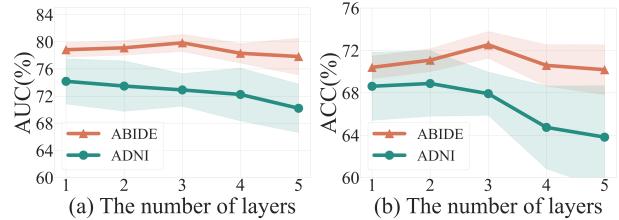
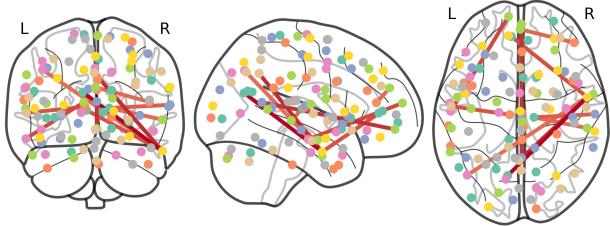


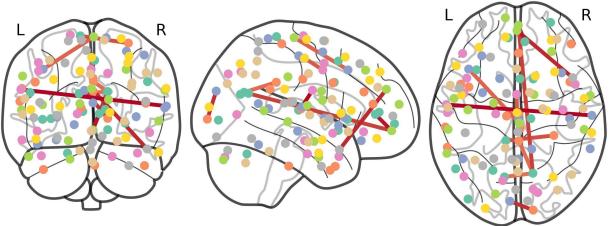
Figure 5. Performance variations for varying  $k$  on two datasets.

Fig. 5. From this figure, it is evident that the proposed BQN achieves stable performance on these two datasets when the parameter  $k$  is within the range  $\{1, 2, 3\}$ . This demonstrates the stability of BQN. Moreover, the experimental results indicate that the model performance exhibits a gradual decline as the number of layers increases. Considering that brain datasets are generally of limited size (on the order of hundreds of samples), the observed performance degradation can be attributed to overfitting, which is likely induced by the excessive number of parameters and the heightened model complexity associated with deeper architectures.

**Case Study.** To interpret the rationality of BQN on the ABIDE dataset, we construct two contrast brain graphs that reflect brain network differences among distinct groups. The first contrast brain graph is derived from the initial brain connectivity matrices, while another contrast brain graph is from the brain network generated by BQN. Specifically, the brain connectivity matrix template of ASD is obtained by averaging functional connectivity matrices within ASD groups,  $A_{\text{Template}}^{\text{ASD}} = \frac{1}{n} \sum_{i=1}^n A_i^{\text{ASD}}$ . Brain template of NC is obtained similarly,  $A_{\text{Template}}^{\text{NC}} = \frac{1}{m} \sum_{i=1}^m A_i^{\text{NC}}$ , where  $m$  denotes the number of normal subjects, and  $n$  represents the number of ASD patients in the ABIDE dataset. The contrast brain graph is then calculated as:  $A_{\text{contrast}} = A_{\text{template}}^{\text{NC}} \ominus A_{\text{Template}}^{\text{ASD}}$ , where  $\ominus$  performs element-wise subtraction on the two input matrices. The learned contrast brain graph is then obtained by performing the same process as the equations above. The only difference is that the output of the last layer of a well-trained BQN is as brain graphs to construct the learnable contrast graph. Notably, the input of the well-trained BQN is a test set, and the phenotype of the subject is based on the result of the classification head of the model.



(a) Contrast brain graph constructed from initial data



(b) Contrast brain graph constructed from the output of BQN

*Figure 6.* Brain network comparison. Red edges denote significant differences in corresponding brain connectivity between the normal population and individuals with Autism Spectrum Disorder.

The visualization of the two contrast graphs, the top-20 edges with the largest weights are selected, is shown in Fig. 6. Firstly, it can be observed that compared to the initial contrast brain graph (subfigure a), the contrast graph learned by BQN is sparser and retains the core focus of abnormal connectivity, such as connectivity between paracentral lobule region and cingulum region. This finding indicates that the BQN learn a more holistic brain functional connectivity and gain ability of learning characteristic representations of brain functional modules, consistent with the theoretical analysis.

Moreover, our model focuses on the abnormal connections of the prefrontal cortex, cingulate, corpus callosum, and parietal as is shown in Fig. 6. Disruption of functional connectivity in these brain regions has been shown to be strongly associated with ASD patients (Assaf et al., 2010; Weng et al., 2010). This indicates that our model possesses a high degree of biological interpretability, rather than merely an improvement in performance.

## 7. Conclusions

This paper observed that the message-passing mechanism, commonly used in brain network analysis, is redundant from both model analysis and experimental investigation. Based on this finding, BQN is introduced by employing Quadratic Network. BQN is a simple but novel model that adaptively learns brain functional networks while implicitly clustering connectivity between brain regions. Comprehensive experiments on the ABIDE and ADNI datasets demonstrate that

BQN consistently outperforms models based on GNNs and Transformers. We hope this study could offer valuable insights into simpler models that can also achieve remarkable performance in brain disorder disease classification.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. U22B2036, 62376088, 62025604, 92370111, 62272340, 62261136549), in part by the Hebei Natural Science Foundation (No. F2024202047), in part by the National Science Fund for Distinguished Young Scholarship (No. 62025602), in part by the Hebei Yanzhao Golden Platform Talent Gathering Programme Core Talent Project (Education Platform) (HJZD202509), in part by the Post-graduate's Innovation Fund Project of Hebei Province (CXZZBS2025036), in part by the Tencent Foundation, and in part by the XPLORER PRIZE.

## Impact Statement

This paper presents work whose goal is to advance the field of Machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

## References

- Assaf, M., Jagannathan, K., Calhoun, V. D., Miller, L., Stevens, M. C., Sahl, R., O’Boyle, J. G., Schultz, R. T., and Pearlson, G. D. Abnormal functional connectivity of default mode sub-networks in autism spectrum disorder patients. *Neuroimage*, 53(1):247–256, 2010.
- Barabási, A.-L. Network science. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 371(1987):20120375, 2013.
- Bassett, D. S. and Bullmore, E. Small-world brain networks. *The neuroscientist*, 12(6):512–523, 2006.
- Bendich, P., Marron, J. S., Miller, E., Pieloch, A., and Skwerer, S. Persistent homology analysis of brain artery trees. *The annals of applied statistics*, 10(1):198, 2016.
- Bertsekas, D. P. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.
- Bessadok, A., Mahjoub, M. A., and Rekik, I. Graph neural networks in network neuroscience. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(5):5833–5848, 2023.
- Bishop, C. M. and Nasrabadi, N. M. *Pattern recognition and machine learning*, volume 4. Springer, 2006.
- Bondy, J. A. and Murty, U. S. R. *Graph theory*. Springer Publishing Company, Incorporated, 2008.

- Cho, H., Sim, J., Wu, G., and Kim, W. H. Neurodegenerative brain network classification via adaptive diffusion with temporal regularization. In *ICML*, 2024.
- Chung, M. K. *Brain network analysis*. Cambridge University Press, 2019.
- Cui, H., Dai, W., Zhu, Y., Kan, X., Gu, A. A. C., Lukemire, J., Zhan, L., He, L., Guo, Y., and Yang, C. Braingb: a benchmark for brain network analysis with graph neural networks. *IEEE transactions on medical imaging*, 42(2): 493–506, 2022.
- Cui, P., Wang, X., Pei, J., and Zhu, W. A survey on network embedding. *IEEE Trans. Knowl. Data Eng.*, 31(5):833–852, 2019.
- Dosenbach, N. U., Fair, D. A., Miezin, F. M., Cohen, A. L., Wenger, K. K., Dosenbach, R. A., Fox, M. D., Snyder, A. Z., Vincent, J. L., Raichle, M. E., et al. Distinct brain networks for adaptive and stable task control in humans. *Proceedings of the National Academy of Sciences*, 104(26):11073–11078, 2007.
- Dosenbach, N. U., Nardos, B., Cohen, A. L., Fair, D. A., Power, J. D., Church, J. A., Nelson, S. M., Wig, G. S., Vogel, A. C., Lessov-Schlaggar, C. N., et al. Prediction of individual brain maturity using fmri. *Science*, 329(5997): 1358–1361, 2010.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.
- Fan, F., Cong, W., and Wang, G. A new type of neurons for machine learning. *International journal for numerical methods in biomedical engineering*, 34(2):e2920, 2018.
- Fan, F., Xiong, J., and Wang, G. Universal approximation with quadratic deep networks. *Neural Networks*, 124: 383–392, 2020.
- Fan, F., Li, M., Wang, F., Lai, R., and Wang, G. On expressivity and trainability of quadratic networks. *IEEE Trans. Neural Networks Learn. Syst.*, 36(1):1228–1242, 2025.
- Fornito, A., Zalesky, A., and Bullmore, E. *Fundamentals of brain network analysis*. Academic press, 2016.
- Garcia, J. O., Ashourvan, A., Muldoon, S., Vettell, J. M., and Bassett, D. S. Applications of community detection techniques to brain graphs: Algorithmic considerations and implications for neural function. *Proceedings of the IEEE*, 106(5):846–867, 2018.
- Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural message passing for quantum chemistry. In *ICML*, pp. 1263–1272. PMLR, 2017.
- Kan, X., Cui, H., Lukemire, J., Guo, Y., and Yang, C. Fbnetgen: Task-aware gnn-based fmri analysis via functional brain network generation. In *MIDL*, pp. 618–637. PMLR, 2022a.
- Kan, X., Dai, W., Cui, H., Zhang, Z., Guo, Y., and Yang, C. Brain network transformer. In *NeurIPS*, 2022b.
- Keshun, Y., Puzhou, W., and Yingkui, G. Towards efficient and interpretive rolling bearing fault diagnosis via quadratic neural network with bi-lstm. *IEEE Internet of Things Journal*, 2024.
- Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.
- Klicpera, J., Bojchevski, A., and Günnemann, S. Predict then propagate: Graph neural networks meet personalized pagerank. In *ICLR*, 2019.
- Kreuzer, D., Beaini, D., Hamilton, W., Létourneau, V., and Tossou, P. Rethinking graph transformers with spectral attention. *NeurIPS*, 34:21618–21629, 2021.
- Li, Q., Han, Z., and Wu, X. Deeper insights into graph convolutional networks for semi-supervised learning. In *AAAI*, pp. 3538–3545, 2018.
- Li, X., Zhou, Y., Dvornek, N. C., Zhang, M., Gao, S., Zhuang, J., Scheinost, D., Staib, L. H., Ventola, P., and Duncan, J. S. Braingnn: Interpretable brain graph neural network for fmri analysis. *Medical Image Analysis*, 74: 102233, 2021.
- Liao, J.-X., Dong, H.-C., Sun, Z.-Q., Sun, J., Zhang, S., and Fan, F.-L. Attention-embedded quadratic network (qttention) for effective and interpretable bearing fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 72:1–13, 2023.
- Luo, X., Liu, Z., Jin, L., Zhou, Y., and Zhou, M. Symmetric nonnegative matrix factorization-based community detection models and their convergence analysis. *IEEE Trans. Neural Networks Learn. Syst.*, 33(3):1203–1215, 2022.
- Meunier, D., Lambiotte, R., Fornito, A., Ersche, K., and Bullmore, E. T. Hierarchical modularity in human brain functional networks. *Frontiers in neuroinformatics*, 3: 571, 2009.
- Peng, C., Huang, Y., Dong, Q., Yu, S., Xia, F., Zhang, C., and Jin, Y. Biologically plausible brain graph transformer. *arXiv preprint arXiv:2502.08958*, 2025.

- Rampásek, L., Galkin, M., Dwivedi, V. P., Luu, A. T., Wolf, G., and Beaini, D. Recipe for a general, powerful, scalable graph transformer. In *NeurIPS*, 2022.
- Shehzad, A., Zhang, D., Yu, S., Abid, S., Kumar, D. K., and Xia, F. Multiscale graph transformer for brain disorder diagnosis. *IEEE Transactions on Consumer Electronics*, 2025.
- Sporns, O. and Kötter, R. Motifs in brain networks. *PLoS biology*, 2(11):e369, 2004.
- Topping, J., Giovanni, F. D., Chamberlain, B. P., Dong, X., and Bronstein, M. M. Understanding over-squashing and bottlenecks on graphs via curvature. In *ICLR*, 2022.
- Váša, F. and Mišić, B. Null models in network neuroscience. *Nature Reviews Neuroscience*, 23(8):493–504, 2022.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *NeurIPS*, pp. 5998–6008, 2017.
- Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. *ICLR*, 2017.
- Wang, X., Jin, D., Cao, X., Yang, L., and Zhang, W. Semantic community identification in large attribute networks. In *AAAI*, pp. 265–271, 2016.
- Weng, S.-J., Wiggins, J. L., Peltier, S. J., Carrasco, M., Risi, S., Lord, C., and Monk, C. S. Alterations of resting state functional connectivity in the default network in adolescents with autism spectrum disorders. *Brain research*, 1313:202–214, 2010.
- Wu, F., Jr., A. H. S., Zhang, T., Fifty, C., Yu, T., and Weinberger, K. Q. Simplifying graph convolutional networks. In *ICML*, pp. 6861–6871, 2019.
- Wu, Z., Jain, P., Wright, M., Mirhoseini, A., Gonzalez, J. E., and Stoica, I. Representing long-range context for graph neural networks with global attention. *NeurIPS*, 34:13266–13279, 2021a.
- Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. A comprehensive survey on graph neural networks. *IEEE Trans. Neural Networks Learn. Syst.*, 32(1):4–24, 2021b.
- Xia, F., Sun, K., Yu, S., Aziz, A., Wan, L., Pan, S., and Liu, H. Graph learning: A survey. *IEEE Transactions on Artificial Intelligence*, 2(2):109–127, 2021.
- Xu, J., Yang, Y., Huang, D., Gururajapathy, S. S., Ke, Y., Qiao, M., Wang, A., Kumar, H., McGeown, J., and Kwon, E. Data-driven network neuroscience: On data collection and benchmark. *NeurIPS*, 36:21841–21856, 2023.
- Xu, J., Bian, Q., Li, X., Zhang, A., Ke, Y., Qiao, M., Zhang, W., Sim, W. K. J., and Gulyás, B. Contrastive graph pooling for explainable classification of brain networks. *IEEE Transactions on Medical Imaging*, 2024.
- Yang, J. and Leskovec, J. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *WSDM*, pp. 587–596, 2013.
- Yang, L., Kang, L., Zhang, Q., Li, M., He, D., Wang, Z., Wang, C., Cao, X., Guo, Y., et al. Open: Orthogonal propagation with ego-network modeling. *NeurIPS*, 35: 9249–9261, 2022.
- Yang, L., Wang, J., He, D., Wang, C., Cao, X., Niu, B., and Wang, Z. Graph reciprocal neural networks by abstracting node as attribute. In *ICDM*, pp. 1463–1468. IEEE, 2023.
- Ying, C., Cai, T., Luo, S., Zheng, S., Ke, G., He, D., Shen, Y., and Liu, T. Do transformers really perform badly for graph representation? In *NeurIPS*, pp. 28877–28888, 2021a.
- Ying, C., Cai, T., Luo, S., Zheng, S., Ke, G., He, D., Shen, Y., and Liu, T.-Y. Do transformers really perform badly for graph representation? *NeurIPS*, 34:28877–28888, 2021b.
- Yu, S., Jin, S., Li, M., Sarwar, T., and Xia, F. Long-range brain graph transformer. In *NeurIPS*, 2024.
- Zhai, S., Talbott, W., Srivastava, N., Huang, C., Goh, H., Zhang, R., and Susskind, J. An attention free transformer. *arXiv preprint arXiv:2105.14103*, 2021.
- Zhang, S., Chen, X., Shen, X., Ren, B., Yu, Z., Yang, H., Jiang, X., Shen, D., Zhou, Y., and Zhang, X.-Y. A-gcl: Adversarial graph contrastive learning for fmri analysis to diagnose neurodevelopmental disorders. *Medical Image Analysis*, 90:102932, 2023.
- Zhuo, J., Cui, C., Fu, K., Niu, B., He, D., Guo, Y., Wang, Z., Wang, C., Cao, X., and Yang, L. Propagation is all you need: A new framework for representation learning and classifier training on graphs. In *ACM MM*, pp. 481–489, 2023.
- Zhuo, J., Liu, Y., Lu, Y., Ma, Z., Fu, K., Wang, C., Guo, Y., Wang, Z., Cao, X., and Yang, L. Dualformer: Dual graph transformer. In *ICLR*, 2025.

In the appendix, the presented contents are as follows:

- A: Further elaboration on GNNs and Transformers.
- B: More metrics' analysis.
- C: Theoretical and application progress of quadratic networks.
- D: Experiments on the clustering characteristics of node representations.
- E: Further comparative experiments of BQN.
- F: Uniform configurations.

## A. Further Elaboration on GNNs and Transformers

To better illustrate the message passing mechanism of GNNs, we adopt GCN, SGC (Wu et al., 2019), and APPNP (Klicpera et al., 2019) as examples. These models employ the weighted average function to implement Eq. (1) and Eq. (2) as:

$$\begin{aligned} GCN : \quad & \mathbf{H}^l = \sigma(\tilde{\mathbf{A}}\mathbf{H}^{l-1}\mathbf{W}^l), \\ SGC : \quad & \mathbf{H}^l = \tilde{\mathbf{A}}\mathbf{H}^{l-1} = \tilde{\mathbf{A}}^l\mathbf{X}, \\ APPNP : \quad & \mathbf{H}^l = (1 - \alpha)\tilde{\mathbf{A}}\mathbf{H}^{l-1} + \alpha\mathbf{X}, \end{aligned} \quad (16)$$

where  $\sigma$  denotes the non-linear activation function (e.g., ReLU( $\cdot$ )),  $\alpha$  is the balancing hyper-parameter and  $\mathbf{H}^0 = \mathbf{X}$  denotes the initial node attributes.  $\tilde{\mathbf{A}} = (\mathbf{D} + \mathbf{I}_n)^{-\frac{1}{2}}(\mathbf{A} + \mathbf{I}_n)(\mathbf{D} + \mathbf{I}_n)^{-\frac{1}{2}}$  represents the normalized adjacency matrix with self-loop. Above three models achieve enhanced neighborhood aggregation and residual combination by employing a meticulously designed message passing function, which significantly improves the model's performance and accuracy in capturing complex patterns and relationships within the data. Recently, GNNs struggle to capture long-range dependencies since stacking multiple layers tends to cause over-smoothing and over-squashing issue (Li et al., 2018; Topping et al., 2022) and leads to the loss of discriminative information.

For the representation learning function Eq. (4) in Transformer Section 3.3, it can be reformulated in token-wise form as

$$\hat{\mathbf{z}}_v = \sum_{u \in \mathcal{V}} \mathbf{S}_{v,u}^{QK} \cdot \mathbf{v}_u = \sum_{u \in \mathcal{V}} \frac{\exp(sim(\mathbf{q}_v, \mathbf{k}_u))}{\sum_{u \in \mathcal{V}} \exp(sim(\mathbf{q}_v, \mathbf{k}_u))} \cdot \mathbf{v}_u, \quad (17)$$

where  $sim(\cdot, \cdot)$  terms the similarity function, which mainly adopts scaled dot-product attention, i.e.,  $sim(\mathbf{Q}, \mathbf{K}) = \mathbf{Q}\mathbf{K}^\top/\sqrt{F}$ . Thus, the attention score matrix  $\mathbf{S}^{QK} \in \mathbb{R}^{N \times N}$  acts as the propagation weights, and the Transformer follows the holistic message-passing scheme.

## B. More Metrics' Analysis

Owing to the restricted number of datasets employed and the absence of consistent trends across multiple metrics, we conducted additional experiments to support the conclusions as presented in Fig. 2, 3 and 5.

To evaluate the performance of the basic classifier, GNN and Transformer (Fig. 2), we further conducted experiments on the ABIDE and ADNI datasets, incorporating Precision and Recall as metrics. The results are presented in Fig. 7. These additional experiments reveal consistent trends across the datasets, supporting the hypothesis that the simple classifier outperforms the basic GNN and Transformer with message-passing strategy.

For Fig. 3 (performance comparison of QNN and GCN) and Fig. 5 (performance variations for varying layers), we adopted the micro-F1 score to provide a more comprehensive evaluation of model performance in the above two experiments. The experimental results are presented in Fig. 8. Subfigure (a) shows that the QNN significantly and consistently outperforms the GCN as the number of layers increases. In subfigure (b), BQN achieves stable performance when the number of layers increases, with the best performance occurring at layer 3 on the ABIDE dataset. In contrast, the results indicate a gradual performance decline as the number of layers increases on the ADNI dataset. These experimental trends align with the findings from the main-body experiments in Fig. 3 and 5.

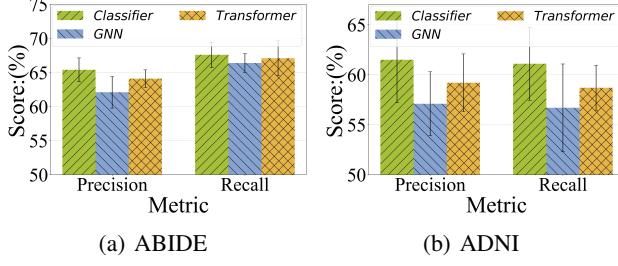


Figure 7. Performances of basic classifier, GNN and Transformer on ABIDE (left) and ADNI (right) datasets according to Precision and Recall. Note that the simple classifier consistently outperforms basic GNN and Transformer with message-passing strategy.

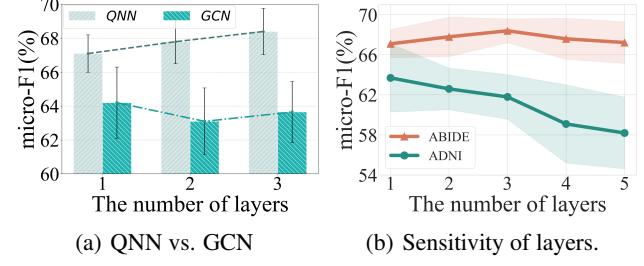


Figure 8. Performance comparison of QNN (Hadamard product) and GCN (matrix product) on ABIDE dataset according to micro-F1 (left), and performance variations for varying k on ABIDE and ADNI datasets according to micro-F1 (right).

## C. Theoretical and Application Progress of Quadratic Networks

The Quadratic Neural Network (QNN) was proposed by authors in (Fan et al., 2018). Subsequent researches have further explored the expressive capabilities of quadratic networks. For instance, (Fan et al., 2020) presented four theorems highlighting the advantages of QNNs over conventional networks in expressive efficiency, unique representation, compact architecture and computational capacity. More recently, (Fan et al., 2025) demonstrated that QNN can implement XOR logic operations and exhibit superior universal approximation properties compared to linear functions.

The findings mentioned above have inspired numerous application-oriented studies. AFT (Zhai et al., 2021) introduces an efficient alternative quadratic function to traditional Transformers that eliminates the need for dot product self-attention, demonstrating competitive performance on all the benchmarks, while providing excellent efficiency at the same time. QCNN (Liao et al., 2023) proposes a novel quadratic convolutional neural network for bearing fault diagnosis, incorporating a derived attention mechanism termed qttention that enhances interpretability and effectiveness. QNN-Bi-LSTM (Keshun et al., 2024) utilize a hybrid model combining a quadratic neural network (QNN) with a bidirectional long short-term memory network (Bi-LSTM) for efficient and interpretable rolling bearing fault diagnosis. Graph Reciprocal Network (GRN) (Yang et al., 2023) treats node attributes and topology of graph as reciprocal elements by regarding nodes as another kind of attribute and employing a novel representation scheme based on Quadratic Networks to achieve fine-grained element-wise product representations.

Previous researches have shown the broad applicability of Quadratic Networks across various domains. However, no studies have yet applied Quadratic Networks-based methods to brain network analysis. Our work fills this gap by showing that Eq. (9) of the quadratic function reveals a clustering effect in brain functional regions. The competitive performance of the Brain Quadratic Network (BQN) presented in Section 6 underscores the potential of Quadratic Networks for brain network analysis and validates its reasonable application in this context.

Table 3. Clustering performance with increasing number of layers

Metric	init	layer_1	layer_2	layer_3
SC↑	0.004	0.179	0.250	0.327
CH↑	5.746	12.357	18.877	22.528
DB↓	4.974	3.434	2.674	1.795

## D. Experiments on the Clustering Characteristics of Node Representations

The node representation obtained through iterative learning of the multi-layers of Eq. (9), its clustering characteristic has been proved in Section 5.2 theoretically. In this section, we further experimentally validate its clustering property. Following the segmentation criteria outlined in (Dosenbach et al., 2007) and (Dosenbach et al., 2010), the brain is segmented into corresponding functional regions. To assess the clustering characteristics of BQN, we employ three standard cluster metrics: the Silhouette Coefficient (SC), Calinski-Harabasz Index (CH) and Davies-Binould Index (DB). Results are obtained from both the original data and the outputs of three well trained BQN models, using 1 layer, 2 layers and 3 layers respectively,

which are presented in Tab. 3. It can be observed from the table that the proposed BQN effectively captures the clustering properties of functional brain regions. Notably, the results reveal that clustering performance increases as the number of layers increase when the number of BQN layers is within {1, 2, 3}, which is consistent with Section 5.3.

## E. Further Comparative Experiments of BQN

Considering the limited number of datasets and related studies used in main body, further comparative experiments are employed on ADHD-200 and PPMI datasets. The details of the two datasets and studies we selected are as follows:

**Datasets.** (1) *Attention Deficit Hyperactivity Disorder (ADHD-200)* dataset is a collaboration of 8 international imaging sites that has aggregated and openly sharing neuroimaging data from children and adolescents diagnosed with ADHD and typically developing controls. The dataset used in this paper includes 459 subjects, comprising 230 typically normal individuals and 229 ADHD patients. (2) *Parkinson’s Progression Markers Initiative (PPMI)* dataset aims to identify biological markers of Parkinson’s risk, onset and progression. Similar to ABIDE, ADNI and ADHD-200, PPMI is also a multi-site dataset. For this paper, the dataset includes 15 normal controls (NC), 67 prodromal individuals, and 113 Parkinson’s disease (PD) patients.

The ROI definition in ADHD-200 dataset is based on Craddock 200 atlas, brain graphs are constructed by computing the Pearson correlation coefficients. Specifically, pearson matrices serve as node attributes and brain functional connectivity matrices. For PPMI dataset, ROI definition is based on AAL 116 atlas and node features, edge weights were preprocessed by authors in (Xu et al., 2023).

**Baseline Methods.** Three supplemental baselines are appended to comparison experiments, including A-GCL (Zhang et al., 2023), AGT (Cho et al., 2024) and BrainMGT (Shehzad et al., 2025). Furthermore, the ALTER employed in the primary text was utilized as a baseline for this comparison experiment, thereby providing stronger contrast. The first two studies are GNNs specially designed for Brain Network Analysis, while BrainMGT and ALTER is based on Transformer.

**Metrics.** AUC, ACC, SEN and SPE are used to evaluate models performance on ADHD-200 dataset, consistent with the ABIDE and ADNI datasets. For PPMI dataset, we utilize Accuracy, Precision, Recall and Specificity to assess the classification performance of all baselines. **Besides**, model efficiency comparisons, e.g., training time-Time(s), are included on both two datasets.

**Implementation Details.** For the above two datasets, random partitions were used with a ratio of 7 : 1 : 2. The training process was configured for 200 epochs, with the batch size within the range of {4, 8, 16}. **Notably, to ensure a fair assessment of the baselines in conjunction with BQN, the epoch exhibiting the lowest loss on the validation set is selected for performance evaluation on the test set across all models.** The Adam optimizer, number of layers, and dropout rate were consistent with the configuration selection used for the ABIDE and ADNI datasets in Section 6.1. Moreover, training times reported in Tab. 4 are the total times of training 100 epochs in full batch mode on each dataset.

**Experimental Results.** The performance comparison between BQN and the baseline models on the ADHD-200 and PPMI datasets are presented in Tab. 4. Note that AGT is specifically developed for multi-class classification tasks, whereas ADHD-200 is a binary classification task. Thus, performance comparison with AGT is limited on PPMI dataset, which has three classes. The results reported in the table indicate that BQN achieved optimal performance in three of the four classification metrics on the ADHD-200 dataset, demonstrating its superiority. On the PPMI dataset, BQN is not superior to AGT but still shows competitive performance. **Furthermore, in terms of model efficiency, our approach exhibits the shortest training time across both datasets.** Above experimental results highlight the promising potential of our model.

Table 4. Performance comparison on ADHD-200 and PPMI. The best and the runner-up models is in bold and underlined, respectively.

Model	ADHD-200					PPMI				
	AUC↑	ACC↑	SEN↑	SPE↑	Time(s)↓	Accuracy↑	Precision↑	Recall↑	Specificity↑	Time(s)↓
AGT	-	-	-	-	-	<b><u>77.69±2.81</u></b>	<b><u>56.42±2.46</u></b>	73.69±3.50	<b><u>79.37±2.27</u></b>	11.09
A-GCL	<u>74.78±4.39</u>	<u>73.11±4.30</u>	<u>72.04±4.68</u>	<u>73.08±4.10</u>	19.71	73.65±3.19	48.99±3.71	<u>70.73±3.01</u>	75.67±2.19	14.12
ALTER	<u>83.16±1.61</u>	<u>73.48±1.34</u>	<u>74.88±6.85</u>	<u>72.20±5.82</u>	<u>17.38</u>	73.91±3.13	50.29±3.01	<u>71.81±5.64</u>	<u>77.36±4.22</u>	11.55
BrainMGT	<u>78.33±1.10</u>	<u>71.27±2.26</u>	<u>71.09±5.63</u>	<b><u>73.26±3.49</u></b>	26.35	70.61±2.37	47.10±2.26	68.45±4.81	74.29±6.40	17.55
BQN (Ours)	<b><u>83.34±1.13</u></b>	<b><u>75.68±1.95</u></b>	<b><u>79.73±2.27</u></b>	71.63±4.87	<b><u>6.56</u></b>	74.26±3.95	52.30±3.75	<b><u>73.93±4.27</u></b>	76.25±3.56	<b><u>4.79</u></b>

## F. Uniform Configurations

Existing GNNs and Transformers for brain networks are very different in data processing and model selection, which makes it difficult to fairly assess their performance. To alleviate this difficulty, we unifies these settings and compares methods in a fair manner. Unifying settings of our work can be concluded as two aspects:

- **Model Selection.** We unify early stopping criteria as the lowest loss on the validation set, which is consistent with BrainGNN and many traditional GNNs and Graph transformers.
- **Data Processing.** We employed a uniform preprocessing approach and brain network construction method for all models on each dataset. Specifically, for the ABIDE dataset, our preprocessing and brain network construction were aligned with those used by BrainNETTF, ALTER and BioBGT. While for the ADNI dataset, we maintained consistency with ALTER’s preprocessing and brain network construction methods. For the ADHD-200 dataset, our preprocessing and brain network construction were consistent with BioBGT’s methods. Finally for the PPMI dataset, the node attributes and edge weights were preprocessed by authors in ([Xu et al., 2023](#)).