When Language Models Meet NeuroGraphs: Exploring Enhanced Agentic LLM Framework Towards Brain Network Analysis

Jiaxing Li[1*] **Rui Dong**[1*] **Muyao Tang**[1] **Youyong Kong**[1†]
[1]School of Computer Science and Engineering, Southeast University
jiaxing_li@seu.edu.cn, 1436598249@qq.com, mytang@seu.edu.cn
kongyouyong@seu.edu.cn

Abstract

Brain network analysis is crucial for understanding cognition and neurological disorders, yet existing deep learning methods mainly treat connectome analysis as a graph-to-logit classification problem, offering limited explanatory reasoning. Large language models (LLMs) provide a promising interface for knowledge-intensive scientific analysis, but directly applying general-purpose LLMs to brain networks remains challenging due to the structure-language gap, limited neuroscience grounding, and overconfident positive predictions. In this paper, we propose **BrainAgent**, an agentic LLM framework for knowledge-enhanced brain network analysis. BrainAgent reformulates connectome classification as an iterative process of topology-aware understanding, external retrieval, reasoning, and reflection. Specifically, it first converts raw brain networks into compact multi-level structural descriptions through brain-specific analysis tools, then retrieves relevant neuroscience knowledge and task-specific cases to ground the reasoning process, and finally generates structured predictions with reflective verification. Experiments on four public rs-fMRI datasets show that BrainAgent consistently improves different closed-source and open-source LLM backbones over direct prompting and standard reasoning baselines. Further ablation and interpretability analyses demonstrate the effectiveness of each component and show that BrainAgent produces more comprehensive, multi-level, and verifiable explanations.These results indicate that agentic LLMs provide a practical route toward interpretable and knowledge-grounded brain network analysis.

1 Introduction

Brain network analysis provides a principled way to study human cognition and neurological disorders by representing neuroimaging measurements as graphs, where nodes denote brain regions and edges encode structural or functional interactions [2]. In particular, resting-state functional MRI (rs-fMRI) enables the construction of functional connectomes and has been widely used for disease-related classification and biomarker discovery. However, brain network analysis is not merely a prediction problem. For clinical and scientific use, a model is expected to explain *why* certain topological patterns are associated with a disorder-related label, and how such evidence is grounded in known neuroscience knowledge.

Existing methods mainly formulate brain network analysis as a supervised graph classification task. Graph neural networks (GNNs) [21, 42, 26, 28] and standardized GNN benchmarks for brain

*Equal contribution. Code can be available at https://github.com/KamonRiderDR/BrainAgent.
†Corresponding author.

Preprint.

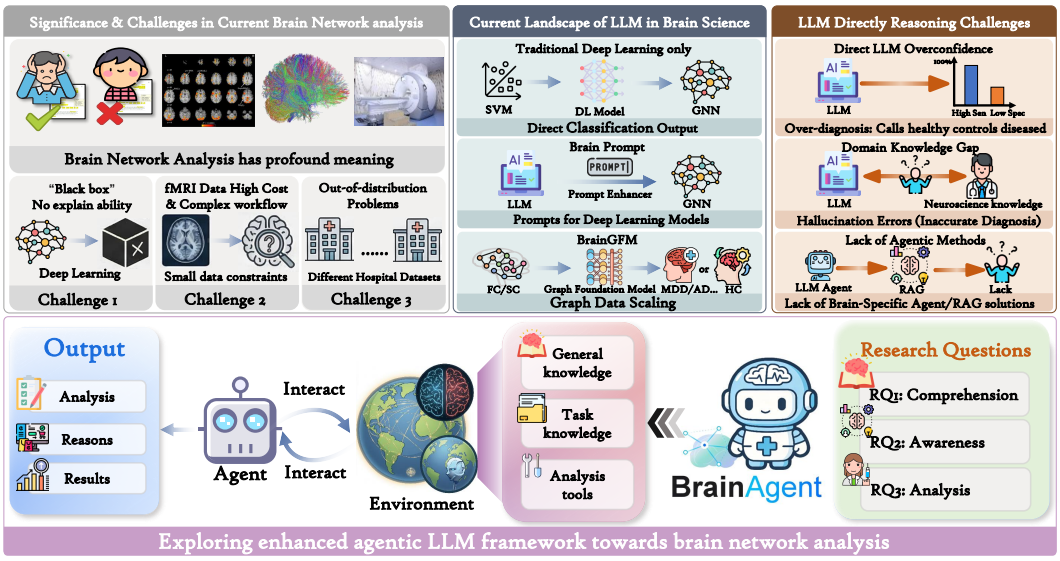


Figure 1: Illustration of the motivation and overall framework of **BrainAgent**. We position brain network analysis as a knowledge-intensive, agentic reasoning problem. BrainAgent addresses this problem through topology-aware graph understanding, tool-augmented interaction, neuroscience knowledge retrieval, task-specific case retrieval, and reflective analysis.

networks [5] have achieved promising performance by learning discriminative graph representations. Nevertheless, this representation-centric paradigm has intrinsic limitations (Figure 1). First, most models map a brain graph to logits through a fixed forward pass, providing limited explanatory reasoning beyond post-hoc saliency or region visualization. Second, rs-fMRI data are typically expensive to acquire, noisy, small-scale, and highly sensitive to preprocessing pipelines, sites, scanners, and cohort differences. As a result, models trained on a fixed dataset often struggle to generalize across heterogeneous clinical scenarios.

Large language models (LLMs) have recently been applied to knowledge-intensive tasks in specialized domains, including medicine [29, 10], e-commerce risk management [30], and finance [46]. In brain science, LLMs are appealing because they can serve as interfaces to heterogeneous textual knowledge, including neuroscience concepts, clinical criteria, and prior case descriptions. However, existing LLM-based brain network studies mostly follow an *LLM-as-enhancer* paradigm. For example, BrainPrompt uses LLM-generated multi-level prompts to enhance conventional GNNs [48], while BrainGFM transfers foundation-model ideas to graph pre-training via graph masked autoencoding and contrastive learning [45]. These approaches improve graph representation learning, but they do not fully exploit LLMs as interactive reasoning engines that can directly analyze a subject-level brain graph, seek external evidence, and produce structured explanations. A natural question is whether we can directly use an LLM as a predictor and analyst for brain networks. Our preliminary observations suggest that this is non-trivial. Directly prompting a general-purpose LLM with serialized connectome data often leads to three failure modes. First, there exists a *structure-language gap*: a brain network is a non-Euclidean graph with global topology, community organization, and region-level semantics, whereas an LLM receives a sequential text input. Naively linearizing edges produces long and sparse contexts that are difficult for the model to interpret. Second, there exists a *domain knowledge gap*: general-purpose LLMs are not explicitly grounded in subject-specific neuroimaging evidence and may generate biologically unsupported explanations. Third, we observe a *positive prediction bias*: directly prompted LLMs tend to over-predict disorder-related classes, resulting in high sensitivity but low precision or specificity. Such overconfident behavior is particularly problematic for medical screening scenarios.

Agentic LLM systems provide a promising direction to address these limitations. Retrieval-augmented generation connects parametric LLMs with external non-parametric memory [24], while tool-using and reasoning-acting frameworks allow LLMs to interact with external environments and APIs [51, 36, 50]. However, directly applying generic agent frameworks to brain network analysis is insufficient. Unlike open-domain question answering, the evidence required here is graph-structured, subject-

specific, and biologically constrained. The agent must first transform a connectome into a topology-grounded reasoning state, then retrieve relevant neuroscience knowledge and comparable cases, and finally produce a calibrated decision with explicit analytical evidence.

To this end, we propose **BrainAgent**, a *brain-specific agentic LLM* framework for knowledge-enhanced brain network analysis. As shown in Figure 1, BrainAgent reformulates brain connectome classification as an iterative process of *understanding*, *retrieval*, *reasoning*, and *reflection*. Given an rs-fMRI brain graph, BrainAgent first invokes brain-specific analytical tools to extract multi-level topological evidence, including region-level, subgraph-level, and global graph features. It then retrieves relevant neuroscience knowledge and task-specific cases through heterogeneous retrieval modules. Finally, the LLM integrates graph evidence, retrieved knowledge, and case memory to generate both a prediction and a structured rationale, followed by a reflection step to mitigate overconfident or inconsistent decisions. Our contributions are summarized as follows:

- We formulate brain network analysis as a knowledge-intensive agentic reasoning problem, where LLMs analyze textualized connectomes with external tools and evidence.
- We propose **BrainAgent**, which integrates topology-aware graph understanding, neuroscience knowledge retrieval, case-level retrieval, and reflection into a unified inference-time framework.
- Experiments on multiple rs-fMRI datasets show that BrainAgent improves prediction, reduces positive prediction bias, and produces more interpretable reasoning.

2 BrainAgent

2.1 Overall Framework

The overall framework of **BrainAgent** is shown in Figure 2. Given an input brain network, BrainAgent first performs an *understanding* stage to extract high-information-density graph descriptors from sparse and verbose raw graph data. This stage converts the original brain network into multi-level structural descriptions, including region-level, subgraph-level, and graph-level features. Compared with directly feeding long edge lists into the LLM, this design reduces token consumption and helps the model better capture brain topology. After understanding, BrainAgent enters a multi-round *agentic iteration* stage. BrainAgent follows a brain-specific *Think–Report–Action–Observation* loop. At each round, the LLM first reasons over the current context, then generates a concise report to summarize key historical information, selects an external tool through a router, and incorporates the returned observation into the next reasoning step. The available tools include brain-network analysis functions, neuroscience knowledge retrieval, and task-specific case retrieval. Finally, BrainAgent produces the final *analysis*, including the predicted label, supporting rationale, and confidence score. To mitigate the overconfidence bias of direct LLM prediction, BrainAgent further uses a reflection module to verify whether the prediction is consistent with graph evidence, retrieved neuroscience knowledge, and similar historical cases.

2.2 Topology-Aware Graph Understanding

The first step of BrainAgent is to convert the raw brain network into an LLM-readable textual representation. Since the values in a functional brain network naturally indicate connection strengths between brain regions, we represent the brain graph as a sequence of triplets in the form of `(src, dst, val)`, where `src` and `dst` denote two brain regions and `val` denotes their connection strength. To better preserve structural information in the sequential input, we sort the triplets according to the degree of the source brain region, so that structurally important regions appear earlier in the context. However, directly using the serialized graph as input may lead to overly long contexts and unreliable reasoning. To address this, BrainAgent invokes analysis functions to extract compact structural descriptors from the original graph. Specifically, for a brain graph $\mathcal{G}$, its serialized textual representation is denoted as $T_{\mathcal{G}}$, and the analysis functions generate corresponding analytical descriptions $T'_{\mathcal{G}}$:

$$\mathcal{G} \rightarrow T_{\mathcal{G}} \xrightarrow{\text{Analysis Functions}} T'_{\mathcal{G}}. \tag{1}$$

The analysis functions include two types of measurements. The first type contains general graph properties, such as degree, density, clustering coefficient, and transitivity. The second type con-

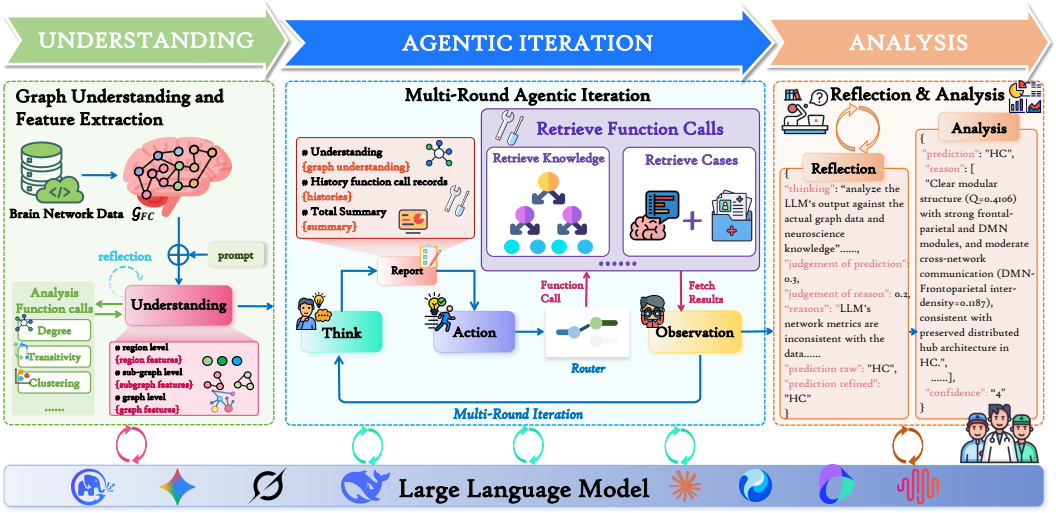


Figure 2: Overall framework of BrainAgent: an understanding module that converts raw brain-network data into compact multi-level structural features, a multi-round agentic iteration module based on a *Think–Report–Action–Observation* loop with analysis tools and external retrieval, and an analysis stage with reflection to verify and refine the final prediction and rationale.

tains brain-network-related properties, such as small-worldness, global efficiency, and community organization. The serialized graph and the analytical descriptions are then jointly fed into the LLM:

$$T_{\text{analysis}} = F_\theta\left(T_{\mathcal{G}}, T'_{\mathcal{G}}; P\right), \tag{2}$$

where P denotes the prompt for graph understanding, and T_{analysis} denotes the generated topology-aware analysis. Here, F_θ is the LLM-based agentic framework defined in the problem formulation, with θ denoting the parameters of the backbone LLM and its inference-time configuration. As shown in Figure 2, the output of this stage is organized into three levels: region-level features, subgraph-level features, and graph-level features. Region-level features describe salient brain regions and hub nodes; subgraph-level features summarize community-level patterns; graph-level features capture whole-brain topological properties. This structured understanding result is used as the initial context for the following agentic reasoning process.

2.3 Hierarchical Augmented Retrieval for Neural Knowledge

Accurate agentic reasoning requires not only tool design but also reliable external knowledge. To enhance the domain awareness of the LLM, BrainAgent introduces the **Hierarchical Augmented Retrieval for Neural Knowledge (HARK)** module. Given the current reasoning context and prompt, HARK retrieves the top-k relevant neuroscience knowledge items and provides them as external evidence for subsequent reasoning. Detailed construction and implementation are provided in Appendix F.

Neuroscience knowledge base construction. High-quality retrieval depends on a high-quality knowledge base. We construct a neuroscience knowledge base from authoritative manuals and highly cited neuroscience literature. The construction pipeline consists of four steps: raw document collection, knowledge-point extraction, LLM-based filtering, and manual verification. Since BrainAgent is evaluated on multiple brain-network analysis tasks, we organize the knowledge base into task-specific knowledge and task-agnostic knowledge. Specifically, for a task T_i, the corresponding task-related knowledge set is denoted as $\mathcal{K}^{T_i}$, while the general neuroscience knowledge set is denoted as $\mathcal{K}^{G}$.

Hierarchical retrieval. BrainAgent retrieves neuroscience knowledge in a hierarchical manner. Since all brain networks in our experiments are constructed under the same AAL atlas, each task-specific knowledge item can be associated with a set of relevant brain regions. Therefore, HARK first performs coarse-grained retrieval based on anatomical region overlap and then conducts fine-grained retrieval based on semantic similarity in the text embedding space.

Coarse-grained retrieval. For task-specific knowledge, each knowledge item is represented as a key-value pair, where the key is the associated brain-region set and the value is the textual knowledge content. Given a query q and a knowledge item k, we compute their anatomical overlap by IoU:

$$\mathrm{sim}(q,k) = \frac{|R_q \cap R_k|}{|R_q \cup R_k|}, \tag{3}$$

where R_q and R_k denote the brain-region sets involved in the query and the knowledge item, respectively. Knowledge items whose IoU scores exceed a predefined threshold are retained as candidates.

Fine-grained retrieval. After coarse retrieval, HARK further ranks the candidate task-specific knowledge items by semantic similarity in the embedding space and returns the top-k most relevant items. For general knowledge $\mathcal{K}^G$, which is not explicitly associated with specific brain regions, HARK directly performs semantic retrieval and returns the top-k results. The retrieved knowledge is then appended to the agent context as external neuroscience evidence, helping BrainAgent reduce hallucination and bridge the domain gap between general-purpose LLMs and brain network analysis.

2.4 Case Augmented Retrieval via Dual-Modality Recall

In addition to general neuroscience knowledge, BrainAgent also requires task-specific evidence from similar brain-network samples. To this end, we introduce **Case Augmented Retrieval via Dual-Modality Recall** (**CARD**), which retrieves relevant historical cases from the support set and uses their reports as in-context evidence. The full construction of the case memory, retrieval database, and graph encoder is provided in Appendix H and Appendix G.

Case memory construction. For each task t, we construct a support case memory $\mathcal{D}_t$ from the training split. Each case contains three components: the brain graph, the ground-truth label, and an LLM-generated brain-network report. The report summarizes the major graph patterns of each sample, including region-level hubs, subgraph-level organization, and global topological properties. In this way, CARD builds a retrieval database that contains both graph-structured evidence and textual analytical evidence.

Dual-modality recall. For a given input brain graph $\mathcal{G}_i$ and a candidate case $\mathcal{G}_j$ from $\mathcal{D}_t$, CARD measures their similarity from both graph and text modalities. The graph-level similarity $\alpha^{\mathcal{G}}_{i,j}$ is computed based on graph representations extracted by a pre-trained brain graph encoder, while the text-level similarity $\alpha^{\mathcal{T}}_{i,j}$ is computed based on the generated reports. The final retrieval score is defined as:

$$\tilde{\alpha}_{i,j} = \beta \cdot \alpha^{\mathcal{G}}_{i,j} + (1-\beta) \cdot \alpha^{\mathcal{T}}_{i,j}, \tag{4}$$

where β controls the balance between graph-level and text-level matching. In our implementation, β is fixed to 0.5.

Case-enhanced reasoning. Based on the hybrid score $\tilde{\alpha}_{i,j}$, CARD returns the top-k most similar cases and their corresponding reports. These retrieved cases help BrainAgent compare the current sample with previously observed HC or disorder-related patterns. The dual-modality design provides finer-grained matching, since graph representations capture topological similarity while textual reports preserve interpretable analytical cues. The retrieved cases are then appended to the agent context as case-level evidence for subsequent reasoning and reflection.

2.5 Reflection and Final Analysis

After multi-round agentic iteration, BrainAgent generates the final analysis. The output includes three parts: the predicted label, the supporting rationale, and the confidence score. The rationale is required to explicitly refer to graph evidence, retrieved neuroscience knowledge, and similar cases whenever available. To further reduce the overconfidence bias of direct LLM prediction, BrainAgent applies a reflection module after the initial analysis. The reflection module checks whether the predicted label is consistent with the extracted graph features, retrieved knowledge, and case-level evidence. If the initial reasoning contains unsupported claims, inconsistent evidence, or excessive confidence, the LLM is asked to revise the analysis and adjust the final prediction when necessary. The final output follows a structured format:

Table 1: Statistics of brain network analysis datasets

Dataset	Task	Samples	Brain Regions ($\|\mathcal{V}\|$)	Classes	Labels
ABIDE	ASD diagnosis	618	90	2	{HC, ASD}
ADHD	ADHD diagnosis	938	90	2	{HC, ADHD}
HCP	Gender classification	1,039	90	2	{Male, Female}
Rest-meta-MDD	MDD diagnosis	2,165	90	2	{HC, MDD}

Structured Output of BrainAgent

```
{
  "prediction": "HC",
  "reason": [
    "Strong bilateral and homologous coupling indicates preserved interhemispheric symmetry.",
    "Dense mesoscale modules with closed triads suggest coherent local functional integration.",
    "......"
  ],
  "confidence": "3"
}
```

This structured output enables BrainAgent to provide not only a prediction result but also an interpretable reasoning process for brain network analysis.

3 Experiments

3.1 Datasets

We evaluate BrainAgent on four public real-world rs-fMRI brain network datasets, including **ABIDE** [8], **ADHD** [4], **HCP** [41], and **Rest-meta-MDD** [3]. The statistics of these datasets are summarized in Table 1. All brain networks are constructed under the AAL atlas with 90 brain regions, and each dataset corresponds to a binary graph-level classification task. Specifically, ABIDE is used for autism spectrum disorder (ASD) diagnosis, ADHD is used for attention deficit hyperactivity disorder diagnosis, HCP is used for gender classification, and Rest-meta-MDD is used for major depressive disorder (MDD) diagnosis. For each dataset, we randomly select 20% of the samples as the test set and use the remaining samples for retrieval database construction and model-related preparation.

3.2 Experimental Settings

We evaluate BrainAgent on both closed-source and open-source LLMs to examine its general applicability. For closed-source models, we use strong proprietary LLMs, including **DeepSeek v3.2** [7], **Qwen3 Max** [49], and **Gemini 3.1**. For open-source models, we adopt **Qwen3.5-35B** [34], **Qwen3.5-9B** [34], and **Gemma4-26B** as representative backbones. For each backbone LLM, we compare the vanilla direct-prompting baseline with BrainAgent. In addition, for closed-source LLMs, we include two widely used reasoning baselines: **Chain-of-Thought (CoT)** [44], which encourages step-by-step reasoning through prompting, and **Reflection** [37], which asks the model to reconsider and revise its initial prediction. For the model generalization study and ablation analysis, we additionally evaluate recent proprietary LLMs including **GPT 5.3**, **Grok 4**, **Hunyuan 3**, and **Seed 2.0**. For models without public technical papers, we report their official API/model identifiers and access sources in Appendix D. All methods are evaluated under the same data split and input protocol. For API-based models, we use the official inference interfaces with a low temperature to reduce randomness. For open-source models, inference is performed locally with standard decoding settings. No supervised fine-tuning is applied to any LLM. BrainAgent only augments the inference process through topology-aware understanding, external retrieval, tool use, and reflection. [3]

3.3 Evaluation Metrics

Following common evaluation protocols for LLM-based prediction tasks, we evaluate all methods using **Accuracy**, **Recall**, and **Precision**. Accuracy measures the overall proportion of correctly

[3] All LLM model API calls involved in this paper are up to April 2026.

Table 2: Performance comparison of three closed-source LLMs, including DeepSeek v3.2, Qwen3 Max, and Gemini 3.1, on four public brain network analysis datasets.

ABIDE

LLM	Method	Acc.		Recall		Precision	
		@1	*@3*	*@1*	*@3*	*@1*	*@3*
DeepSeek v3.2	Direct	47.15	47.15	100.00	100.00	47.15	47.15
	+ Reflection	51.20	72.78	46.56	76.17	53.34	56.52
	+ CoT	48.06	55.37	82.10	90.19	53.58	53.58
	+ BrainAgent	**58.54**	**86.18**	63.69	82.76	**57.45**	**59.52**
Qwen3 Max	Direct	47.96	47.96	95.51	95.51	50.40	50.40
	+ Reflection	50.12	71.85	48.91	78.73	51.32	63.24
	+ CoT	47.96	47.96	95.51	95.51	50.40	50.40
	+ BrainAgent	**56.91**	**78.86**	62.06	84.48	**53.42**	**64.40**
Gemini 3.1	Direct	59.35	59.35	100.00	100.00	53.70	53.70
	+ Reflection	63.41	69.11	96.55	100.00	56.67	60.42
	+ CoT	59.35	59.35	100.00	100.00	53.70	53.70
	+ BrainAgent	**70.25**	**82.93**	96.55	100.00	**62.22**	**73.42**

ADHD

LLM	Method	Acc.		Recall		Precision	
		@1	*@3*	*@1*	*@3*	*@1*	*@3*
DeepSeek v3.2	Direct	37.97	37.97	100.00	100.00	37.97	37.97
	+ Reflection	49.19	81.14	63.38	77.46	47.91	51.29
	+ CoT	37.97	37.97	100.00	100.00	37.97	37.97
	+ BrainAgent	**61.50**	**86.18**	35.21	82.76	**49.01**	**53.92**
Qwen3 Max	Direct	37.97	37.97	100.00	100.00	37.97	37.97
	+ Reflection	48.66	71.12	77.46	85.24	61.76	64.37
	+ CoT	37.97	37.97	100.00	100.00	37.97	37.97
	+ BrainAgent	**56.91**	**76.12**	63.38	88.73	**61.76**	**67.90**
Gemini 3.1	Direct	51.87	52.41	98.59	100.00	44.03	44.37
	+ Reflection	58.82	68.45	88.73	100.00	47.73	54.62
	+ CoT	53.48	56.68	91.55	100.00	44.52	46.71
	+ BrainAgent	**69.52**	**83.96**	90.14	100.00	**56.14**	**70.30**

HCP

LLM	Method	Acc.		Recall		Precision	
		@1	*@3*	*@1*	*@3*	*@1*	*@3*
DeepSeek v3.2	Direct	45.41	45.41	100.00	100.00	45.41	45.41
	+ Reflection	46.38	75.19	56.84	82.19	45.41	51.90
	+ CoT	45.41	45.41	100.00	100.00	45.41	45.41
	+ BrainAgent	**60.84**	**80.67**	54.68	74.74	**50.17**	**54.21**
Qwen3 Max	Direct	45.41	45.41	100.00	100.00	45.41	45.41
	+ Reflection	49.19	72.84	55.12	80.90	48.91	57.76
	+ CoT	45.41	45.41	100.00	100.00	45.41	45.41
	+ BrainAgent	**55.07**	**80.26**	64.21	87.37	**53.50**	**60.76**
Gemini 3.1	Direct	55.07	55.07	100.00	100.00	50.53	50.53
	+ Reflection	59.90	70.05	100.00	100.00	55.07	64.41
	+ CoT	57.49	62.32	72.32	81.25	53.37	60.51
	+ BrainAgent	**63.68**	**89.60**	61.54	89.25	**55.17**	**88.30**

Rest-meta-MDD

LLM	Method	Acc.		Recall		Precision	
		@1	*@3*	*@1*	*@3*	*@1*	*@3*
DeepSeek v3.2	Direct	53.58	53.58	100.00	100.00	53.58	53.58
	+ Reflection	58.01	77.46	52.38	81.93	62.50	64.66
	+ CoT	53.58	53.58	100.00	100.00	53.58	53.58
	+ BrainAgent	**62.14**	**84.29**	60.67	83.19	**69.98**	**72.62**
Qwen3 Max	Direct	51.73	51.73	81.47	81.47	53.24	53.24
	+ Reflection	59.22	70.21	66.12	84.93	65.20	67.83
	+ CoT	51.73	51.73	81.47	81.47	53.24	53.24
	+ BrainAgent	**68.20**	**80.21**	71.25	86.93	**67.88**	**69.11**
Gemini 3.1	Direct	55.76	56.58	100.00	100.00	54.72	55.24
	+ Reflection	59.82	67.90	97.41	100.00	57.36	62.53
	+ CoT	55.76	56.71	100.00	100.00	54.72	55.37
	+ BrainAgent	**71.36**	**81.99**	86.67	100.00	**67.94**	**74.84**

classified samples, Recall measures the proportion of positive samples that are correctly identified, and Precision measures the proportion of truly positive samples among all samples predicted as positive. For Recall and Precision under *pass@k*, we first aggregate the k predictions into a final label following the same pass criterion, and then compute class-wise Recall and Precision based on the aggregated predictions. For experiments involving multiple stochastic LLM runs, we report *pass@k*, where a sample is considered correctly predicted if at least one of the k generated predictions is correct. In our experiments, we report results with $k = 1$ and $k = 3$ when applicable. These metrics allow us to evaluate both the predictive performance and the over-prediction tendency of LLMs in brain network analysis.

3.4 Performance on Different LLMs

Table 2 reports the results of closed-source LLMs on four public datasets. Overall, direct LLM prompting shows limited reliability for brain network analysis. In many cases, vanilla LLMs tend to predict most samples as positive, leading to very high recall but poor accuracy and precision. This indicates a clear positive prediction bias when general-purpose LLMs are directly applied to serialized brain-network data. CoT provides only marginal improvements, while Reflection is more effective by correcting some initial reasoning errors. In contrast, BrainAgent consistently improves accuracy and precision across different datasets and backbone LLMs. The gains are especially evident under *pass@3*, showing that multi-round tool interaction, knowledge retrieval, case retrieval, and reflection help the LLM produce more reliable and evidence-grounded predictions. Nevertheless, the relatively moderate *pass@1* performance also suggests that training-free LLM agents still have limitations in specialized neuroimaging tasks.

Table 3 further reports the performance of open-source LLMs. In addition to the observations from closed-source models, we find that relatively smaller open-source models can also benefit substantially from BrainAgent. Since BrainAgent is a training-free inference framework, this result suggests that competitive performance can be obtained without relying exclusively on the largest available LLM backbone. Instead, by equipping open-source models with topology-aware understanding, external

Table 3: Performance comparison of three open-source LLMs, including Qwen3.5-35B, Qwen3.5-9B, and Gemma4-26B, on four public brain network analysis datasets.

Methods	ABIDE			ADHD			HCP			Rest-meta-MDD		
	Acc.	Recall	Precision	Acc.	Recall	Precision	Acc.	Recall	Precision	Acc.	Recall	Precision
Qwen3.5-35B	47.97	91.38	47.32	42.25	80.28	37.75	52.17	49.47	47.96	53.12	92.24	53.63
+ BrainAgent	**59.35**	81.03	**54.65**	**58.29**	64.79	**46.46**	**58.94**	61.05	**54.72**	**61.89**	90.09	**59.54**
Qwen3.5-9B	41.46	77.59	43.27	39.25	78.87	36.36	49.51	49.47	45.63	52.66	94.40	53.28
+ BrainAgent	**52.85**	81.03	**50.00**	**51.34**	77.46	**42.31**	**57.49**	34.74	**55.93**	**58.43**	78.02	**58.39**
Gemma4-26B	49.59	62.07	47.37	42.78	66.20	36.15	49.28	64.21	46.21	50.12	72.84	54.28
+ BrainAgent	**60.98**	81.03	**55.95**	**56.68**	76.06	**45.76**	**59.22**	75.53	**53.79**	**60.97**	82.76	**59.81**

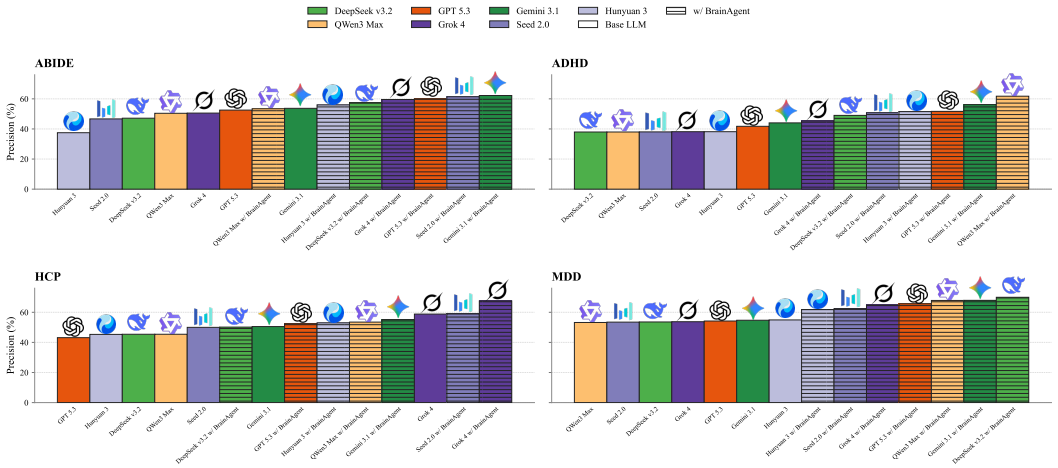


Figure 3: Precision ranking of multiple recent LLMs on four public brain network analysis datasets. BrainAgent consistently improves different backbone LLMs and achieves competitive or leading precision across datasets.

retrieval, and reflection, BrainAgent can effectively inject domain knowledge during inference. This finding also indicates a promising direction for practical deployment: future work can combine BrainAgent with model compression techniques such as distillation and quantization, enabling smaller local models to support privacy-preserving brain network analysis in real clinical scenarios.

3.5 Ablation Studies

To examine the role of each component, we conduct ablation studies on four public datasets with DeepSeek v3.2 and GPT 5.3 as backbone LLMs. Figure 4 reports the *pass@1* accuracy of BrainAgent and its variants. Removing any module reduces performance, confirming that all components contribute to the final result. The understanding module helps convert long raw graph inputs into compact structural descriptions; HARK provides domain knowledge for reducing the neuroscience knowledge gap; CARD brings task-specific case evidence; and reflection verifies the intermediate reasoning and corrects unsupported conclusions. Among them, removing CARD or reflection usually causes larger drops, highlighting the importance of case-level evidence and self-verification in BrainAgent.

3.6 Generalization across Recent LLMs

To further examine whether BrainAgent is robust to different backbone models, we evaluate it on multiple LLMs and rank their precision on four public datasets. As shown in Figure 3, BrainAgent consistently improves different base LLMs and often moves them to the top of the ranking. This indicates that the gains of BrainAgent do not come from a specific model, but from the inference-time enhancement introduced by topology-aware understanding, external retrieval, and reflection. Moreover, even when the original base LLM performs moderately, equipping it with BrainAgent can

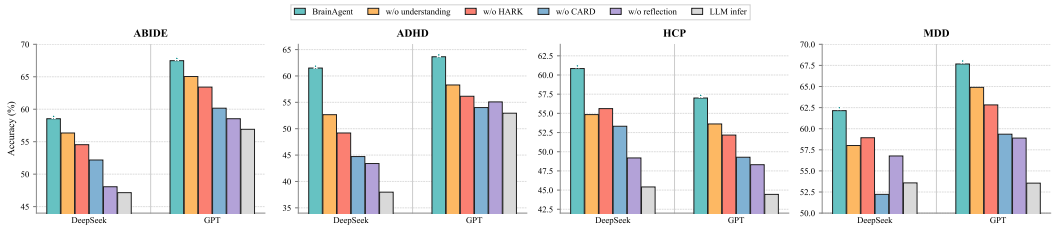
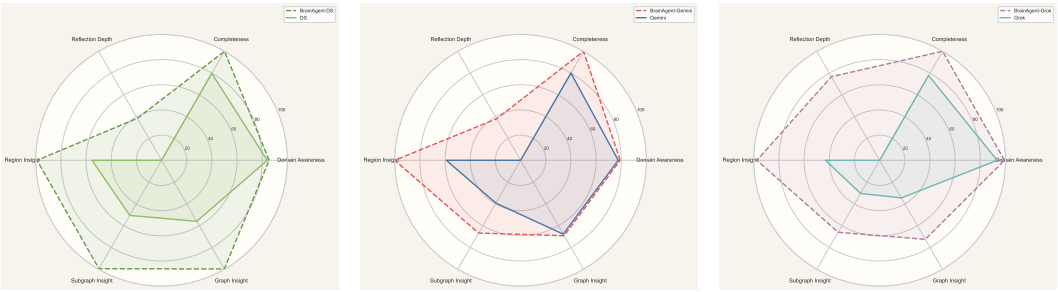


Figure 4: Ablation study of BrainAgent on four public datasets. We report accuracy using DeepSeek v3.2 and GPT 5.3 as backbone LLMs.



(a) DeepSeek (b) Gemini (c) Grok

(d) Hunyuan (e) Qwen (f) Seed

Figure 5: Interpretability comparison between base LLMs and BrainAgent-enhanced LLMs across 6 settings. Each radar plot evaluates explanation quality from 6 dimensions: domain awareness, completeness, reflection depth, region insight, subgraph insight, and graph insight.

substantially improve its precision, suggesting that BrainAgent effectively reduces unreliable positive predictions and provides a general plug-and-play framework for brain network analysis.

3.7 Interpretability Analysis

We evaluate explanation quality from six dimensions: **Domain Awareness**, **Completeness**, **Reflection Depth**, **Region Insight**, **Subgraph Insight**, and **Graph Insight**, as shown in Figure 5. These metrics respectively assess whether the explanation is domain-grounded, complete, self-verified, and supported by region-, subgraph-, and graph-level evidence. The overall explainability score is their average. BrainAgent consistently produces more structured explanations than direct inference by decomposing brain networks into multi-level evidence and verifying the rationale through reflection. Detailed reasoning and reflection examples are provided in Appendix J.

4 Conclusion

We present **BrainAgent**, a topology-aware agentic LLM framework for interpretable brain network analysis. By combining graph understanding, external knowledge retrieval, case retrieval, and reflection, BrainAgent improves LLM-based connectome prediction across multiple rs-fMRI datasets

and backbone models. Experiments further show that BrainAgent mitigates positive prediction bias and produces more structured, evidence-grounded explanations. Future work will explore domain adaptation, better calibration, and extension to multimodal neuroimaging data.

References

[1] Peter Belcak, Greg Heinrich, Shizhe Diao, Yonggan Fu, Xin Dong, Saurav Muralidharan, Yingyan Celine Lin, and Pavlo Molchanov. Small language models are the future of agentic ai. *arXiv preprint arXiv:2506.02153*, 2025.

[2] Ed Bullmore and Olaf Sporns. Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience*, 10(3):186–198, 2009.

[3] Xiao Chen, Bin Lu, Hui-Xian Li, Xue-Ying Li, et al. The direct consortium and the rest-meta-mdd project: towards neuroimaging biomarkers of major depressive disorder. *Psychoradiology*, 2(1):32–42, 03 2022.

[4] ADHD-200 consortium. The adhd-200 consortium: a model to advance the translational potential of neuroimaging in clinical neuroscience. *Frontiers in systems neuroscience*, 6:62, 2012.

[5] Hejie Cui, Wei Dai, Yanqiao Zhu, Xuan Kan, Antonio Aodong Chen Gu, Joshua Lukemire, Liang Zhan, Lifang He, Ying Guo, and Carl Yang. Braingb: A benchmark for brain network analysis with graph neural networks. *IEEE Transactions on Medical Imaging*, 42(2):493–506, 2023.

[6] Hejie Cui, Wei Dai, Yanqiao Zhu, Xiaoxiao Li, Lifang He, and Carl Yang. Interpretable graph neural networks for connectome-based brain disorder analysis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 375–385. Springer, 2022.

[7] DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, et al. Deepseek-v3.2: Pushing the frontier of open large language models, 2025.

[8] Adriana Di Martino, Chao-Gan Yan, Qingyang Li, Erin Denio, Francisco X Castellanos, Kaat Alaerts, Jeffrey S Anderson, Michal Assaf, Susan Y Bookheimer, Mirella Dapretto, et al. The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism. *Molecular psychiatry*, 19(6):659–667, 2014.

[9] Haoyu Dong, Jianbo Zhao, Yuzhang Tian, Junyu Xiong, Shiyu Xia, Mengyu Zhou, Yun Lin, José Cambronero, Yeye He, Shi Han, et al. Spreadsheetllm: Encoding spreadsheets for large language models. *arXiv preprint arXiv:2407.09025*, 2024.

[10] Rui Dong, Zitong Wang, Jiaxing Li, Weihuang Zheng, and Youyong Kong. Bleg: Llm functions as powerful fmri graph-enhancer for brain network analysis, 2026.

[11] Rui Dong, Xiaotong Zhang, Jiaxing Li, Yueying Li, Jiayin Wei, and Youyong Kong. M3d-bfs: a multi-stage dynamic fusion strategy for sample-adaptive multi-modal brain network analysis, 2026.

[12] Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. *arXiv preprint arXiv:2410.10781*, 2024.

[13] Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23, 2021.

[14] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1024–1034, 2017.

[15] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[16] Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37:132876–132907, 2024.

[17] Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. Graphmae: Self-supervised masked graph autoencoders. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pages 594–604, 2022.

[18] Jinlong Hu, Yangmin Huang, Nan Wang, and Shoubin Dong. Brainnpt: Pre-training transformer networks for brain network classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2024.

[19] Songtao Jiang, Tuo Zheng, Yan Zhang, Yeying Jin, Li Yuan, and Zuozhu Liu. Med-moe: Mixture of domain-specific experts for lightweight medical vision-language models. *arXiv preprint arXiv:2404.10237*, 2024.

[20] Xinke Jiang, Rihong Qiu, Yongxin Xu, Wentao Zhang, Yichen Zhu, Ruizhe Zhang, Yuchen Fang, Xu Chu, Junfeng Zhao, and Yasha Wang. Ragraph: A general retrieval-augmented graph learning framework. In *Advances in Neural Information Processing Systems*, 2024.

[21] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[22] Youyong Kong, Wenhan Wang, Xiaoyun Liu, Shuwen Gao, Zhenghua Hou, Chunming Xie, Zhijun Zhang, and Yonggui Yuan. Multi-connectivity representation learning network for major depressive disorder diagnosis. *IEEE Transactions on Medical Imaging*, 42(10):3012–3024, 2023.

[23] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

[24] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.

[25] Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36:28541–28564, 2023.

[26] Xiaoxiao Li, Yuan Zhou, Nicha C. Dvornek, Muhan Zhang, Siyuan Gao, Juntang Zhuang, Dustin Scheinost, Lawrence H. Staib, Pamela Ventola, and James S. Duncan. Braingnn: Interpretable brain graph neural network for fmri analysis. *Medical Image Analysis*, 74:102233, 2021.

[27] Yueying Li, Rui Dong, Xiaoyun Liu, Yonggui Yuan, and Youyong Kong. Neurobridge: Bridging functional and structural brain networks via neural coupling and consistency-guided dynamic graph learning. *Medical Image Analysis*, 110:103993, 2026.

[28] Yueying Li, Jiaxing Li, Yue Zhou, Youyong Kong, and Yonggui Yuan. Neurofield-agl: Neurofield-attentive graph learning on functional connectivity for mental disorder diagnosis. *IEEE Sensors Journal*, 26(5):7730–7742, 2026.

[29] Jie Liu, Wenxuan Wang, Zizhan Ma, Guolin Huang, SU Yihang, Kao-Jung Chang, Haoliang Li, Linlin Shen, Michael Lyu, and Wenting Chen. Medchain: Bridging the gap between LLM agents and clinical practice with interactive sequence. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025.

[30] Nan Lu, Yurong Hu, Jiaquan Fang, Yan Liu, Rui Dong, Yiming Wang, Rui Lin, and Shaoyi Xu. Sherlock: Towards dynamic knowledge adaptation in llm-enhanced e-commerce risk management, 2025.

[31] Renqian Luo, Liai Sun, Yingce Xia, Tao Qin, Sheng Zhang, Hoifung Poon, and Tie-Yan Liu. Biogpt: generative pre-trained transformer for biomedical text generation and mining. *Briefings in bioinformatics*, 23(6):bbac409, 2022.

[32] Nick Mecklenburg, Yiyou Lin, Xiaoxiao Li, Daniel Holstein, Leonardo Nunes, Sara Malvar, Bruno Silva, Ranveer Chandra, Vijay Aski, Pavan Kumar Reddy Yannam, et al. Injecting new knowledge into large language models via supervised fine-tuning. *arXiv preprint arXiv:2404.00213*, 2024.

[33] Liang Peng, Songyue Cai, Zongqian Wu, Huifang Shang, Xiaofeng Zhu, and Xiaoxiao Li. Mmgpl: Multimodal medical data analysis with graph prompt learning. *Medical Image Analysis*, 97:103225, 2024.

[34] Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026.

[35] Khaled Saab, Tao Tu, Wei-Hung Weng, Ryutaro Tanno, David Stutz, Ellery Wulczyn, Fan Zhang, Tim Strother, Chunjong Park, Elahe Vedadi, et al. Capabilities of gemini models in medicine. *arXiv preprint arXiv:2404.18416*, 2024.

[36] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[37] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik R Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[38] Xiao-Wei Song, Zhang-Ye Dong, Xiang-Yu Long, Su-Fang Li, Xi-Nian Zuo, Chao-Zhe Zhu, Yong He, Chao-Gan Yan, and Yu-Feng Zang. Rest: a toolkit for resting-state functional magnetic resonance imaging data processing. *PloS one*, 6(9):e25031, 2011.

[39] Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 491–500, 2024.

[40] Yingzhi Teng, Kai Wu, Jing Liu, Yifan Li, and Xiangyi Teng. Constructing high-order functional connectivity networks with temporal information from fmri data. *IEEE Transactions on Medical Imaging*, 2024.

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

[42] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[43] Xingliang Wang, Zemin Liu, Junxiao Han, and Shuiguang Deng. Rag4gfm: Bridging knowledge gaps in graph foundation models through graph retrieval augmented generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[44] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

[45] Xinxu Wei, kanhao zhao, Yong Jiao, Lifang He, and Yu Zhang. A brain graph foundation model: Pre-training and prompt-tuning across broad atlases and disorders. In *The Fourteenth International Conference on Learning Representations*, 2026.

[46] Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance, 2023.

[47] Jiaxing Xu, Qingtian Bian, Xinhang Li, Aihu Zhang, Yiping Ke, Miao Qiao, Wei Zhang, Wei Khang Jeremy Sim, and Balázs Gulyás. Contrastive graph pooling for explainable classification of brain networks. *IEEE Transactions on Medical Imaging*, 2024.

[48] Jiaxing Xu, Kai He, Yue Tang, Wei Li, Mengcheng Lan, Xia Dong, Yiping Ke, and Mengling Feng. BrainPrompt: Multi-Level Brain Prompt Enhancement for Neurological Condition Identification . In *proceedings of Medical Image Computing and Computer Assisted Intervention – MICCAI 2025*, volume LNCS 15971, pages 172 – 182. Springer Nature Switzerland, September 2025.

[49] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, et al. Qwen3 technical report, 2025.

[50] Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

[51] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2023.

[52] Hongting Ye, Yalu Zheng, Yueying Li, Ke Zhang, Youyong Kong, and Yonggui Yuan. RH-brainFS: Regional heterogeneous multimodal brain networks fusion strategy. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[53] Chao Yi, Dian Chen, Gaoyang Guo, Jiakai Tang, Jian Wu, Jing Yu, Mao Zhang, Sunhao Dai, Wen Chen, Wenjun Yang, et al. Recgpt technical report. *arXiv preprint arXiv:2507.22879*, 2025.

[54] Wenhao Zheng, Liaoyaqi Wang, Dongshen Peng, Hongxia Xu, Yun Li, Hongtu Zhu, Tianfan Fu, and Huaxiu Yao. Multimodal clinical trial outcome prediction with large language models. *arXiv preprint arXiv:2402.06512*, 2024.

A Related Works

A.1 GNN-based Brain network analysis.

Graph neural networks (GNNs) have become a widely used backbone for brain network analysis, as they can model non-Euclidean connectome structures through neighborhood aggregation and message passing [21, 42, 14]. Existing GNN-based studies for neuroimaging analysis can be roughly grouped into three lines. The first line focuses on designing more expressive graph encoders for functional connectivity networks. For example, connectome-specific architectures and graph neural models are designed to capture higher-order topological patterns and long-range dependencies in brain graphs [40, 18]. NeuroField-AGL further introduces a neurofield-attentive graph learning mechanism to model functional connectivity for mental disorder diagnosis [28].

The second line improves interpretability in brain graph learning. BrainGNN learns task-relevant subnetworks and provides interpretable brain-region importance [26], while IBGNN and Contrast-Pool introduce explanation-oriented generators or differentiable pooling mechanisms to identify discriminative substructures [6, 47]. These methods improve the transparency of graph classifiers, but their explanations are usually tied to learned graph representations or post-hoc subnetwork selection.

The third line studies multimodal or multi-connectivity brain graph learning. MCRLN integrates structural connectivity, static functional connectivity, and dynamic functional connectivity for MDD diagnosis through structural-functional and static-dynamic fusion [22]. RH-BrainFS models regional heterogeneity between structural and functional brain networks for multimodal fusion [52], M3D-BFS introduces a sample-adaptive dynamic fusion strategy for multimodal brain network analysis [11], and NeuroBridge bridges functional and structural brain networks through neural coupling and consistency-guided dynamic graph learning [27]. Despite these advances, most existing GNN-based methods remain tied to task-specific training and fixed graph-to-logit prediction. They usually require labeled data for each disease or dataset, provide limited explanations beyond attention weights or post-hoc subnetwork visualization, and are difficult to directly transfer across heterogeneous brain-network tasks. BrainAgent is complementary to these supervised graph learning methods: instead of learning a task-specific classifier, it studies whether an LLM agent can perform training-free, evidence-grounded analysis over subject-level brain graphs.

A.2 LLM-based methods in neuroscience.

Pre-trained language models have been widely adapted to biomedical and clinical scenarios. Early studies such as BioBERT [23] and BioGPT [31] continue pre-training or fine-tuning language models on large-scale biomedical corpora, such as PubMed, to improve domain-specific language understanding. With the development of multimodal LLMs, recent methods further incorporate visual, graph, and structured clinical information. For example, LLaVA-Med builds a vision-language assistant for biomedical image understanding [25], while other studies explore medical multimodal fusion with imaging, video, or structured clinical records [35, 54, 19].

In brain network analysis, LLMs have recently been used to inject external knowledge or semantic information into graph learning. MMGPL introduces graph prompts for fMRI-related multimodal fusion [33]. BLEG uses LLMs as fMRI graph enhancers by generating textual augmentation and aligning language-model representations with GNN representations [10]. BrainPrompt further incorporates ROI-level, subject-level, and disease-level prompts to enhance GNN-based neurological condition identification [48]. BrainGFM studies large-scale brain graph pre-training and prompt-tuning across different atlases and disorders [45]. These methods show that language models and prompt-based knowledge can benefit brain graph representation learning. However, most of them still follow an *LLM-as-enhancer* paradigm, where the LLM mainly provides prompts, textual features, or auxiliary representations for a trained downstream model. In contrast, BrainAgent treats the LLM as an interactive reasoning agent, allowing it to analyze subject-level brain graphs with topology-aware tools, retrieve neuroscience knowledge and similar cases, and verify its own prediction through reflection.

B fMRI Brain Network Construction and Data Preprocessing

The preprocessing pipeline for fMRI data is shown in Figure 6. The Data Preprocessing Assistant for Resting-State Function (DPARSF) MRI toolkit [38] is utilized for fMRI preprocessing. Then the average time series are computed for each brain region with AAL template. Pearson correlation is then calculated as functional connectivity matrix, which denotes the feature matrix for FC (X_{FC}). Its adjacency matrix (A_{FC}) is obtained by thresholding a certain proportional quantization on the functional connectivity matrix.

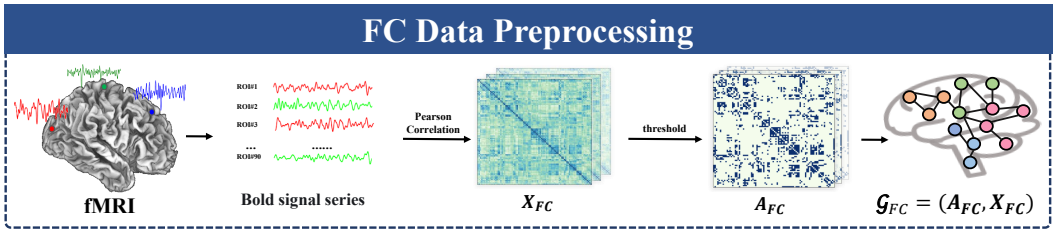


Figure 6: Preprocess of fMRI data and construction for FC dataset.

C Details of Input Format and Function Call Settings

C.1 Graph-Text Transformation

The first step in BrainAgent design is to convert graph data into textual representations before feeding them into LLM. Transforming graph data into formats interpretable by LLMs, as well as enhancing LLMs' capability to comprehend graph topological features, remains a challenging problem [39, 16]. On the other hand, compared to graphs in other domains like social networks or molecular chemistry, brain network data exhibits distinct domain-specific properties. According to the preprocess part B, $\mathcal{E}_{ij}$ represents the connectivity between region i and region j with its value indicating the strength of the connectivity.

Therefore, here we directly represent raw graph data in the form of (`src`, `dst`, `val`) triplets, where `src`, `dst` represent source and target brain regions of each directed edge and `val` stands for connectivity strength. Such transformation ensures complete graph information without information loss. Moreover, we optimize the ordering of triplets to further enhance LLM's perception of raw graph data [9]. Triplets are sorted in descending order according to connectivity and node degrees, where triplets with large `val` and high degree node are placed in the leading positions.

C.2 Analysis Function Call Settings

The basic idea of analysis function calling is to reduce hallucination during inference. For input brain network, LLM tends to conduct certain analysis before final prediction. This process often involves false numerical calculations: They are generated via LLM's next token prediction but not calculated, which often leads to factual errors. Our goal is to provide these analysis results manually in advance. Raw graph data, together with corresponding analysis results are fed into BrainAgent for "Understanding", whose output is utilized as input for agent's subsequent reasoning.

The analysis functions in this paper mainly include following two aspects: (a) Basic attributes of the graph, such as edge density, average node degree, average clustering coefficient, transitivity, etc. (b) Statistical attributes related to brain networks, such as small-world property, whole-brain information transmission efficiency, brain community structure attributes, etc. While calling these analysis function tools, LLM returns functional name and functional description together with analysis results.

Table 4: LLM backbones and access sources used in our experiments.

Model	Provider	Source
DeepSeek v3.2	DeepSeek	Technical report / official API
Qwen3 Max	Alibaba Qwen	Technical report / official API
Gemini 3.1	Google	Official API/model interface
GPT 5.3	OpenAI	Official API/model documentation
Grok 4	xAI	Official API/model interface
Hunyuan 3	Tencent Hunyuan	Official API/model interface
Seed 2.0	ByteDance Seed	Official API/model interface
Qwen3.5-35B	Alibaba Qwen	Official model release
Qwen3.5-9B	Alibaba Qwen	Official model release
Gemma4-26B	Google	Official model card/release

D Experimental Details

D.1 Dataset Details

We conduct experiments on four public rs-fMRI brain network datasets: ABIDE, ADHD, HCP, and Rest-meta-MDD. All datasets are preprocessed under the AAL atlas, resulting in brain networks with 90 regions of interest (ROIs). Each subject is represented as a functional connectivity graph, where nodes denote brain regions and edges denote functional connectivity estimated from rs-fMRI signals. The detailed statistics are reported in Table 1 in the main paper.

ABIDE. The Autism Brain Imaging Data Exchange (ABIDE) dataset [8] is a public multi-site neuroimaging dataset for autism spectrum disorder (ASD) analysis. We use ABIDE for binary classification between healthy controls (HC) and ASD subjects. The dataset contains 618 samples in our experiments.

ADHD. The ADHD dataset [4] is used for attention deficit hyperactivity disorder diagnosis. It contains rs-fMRI samples from both healthy controls and ADHD subjects. We use 938 samples for binary HC/ADHD classification.

HCP. The Human Connectome Project (HCP) dataset [41] is used for gender classification. Compared with disease diagnosis datasets, HCP provides a non-disease classification setting and is used to examine whether BrainAgent can generalize beyond clinical diagnosis tasks. The dataset contains 1,039 samples.

Rest-meta-MDD. Rest-meta-MDD is a large-scale public resting-state fMRI dataset [3] for major depressive disorder (MDD) analysis. It is collected from multiple sites and provides a challenging setting with potential site and cohort heterogeneity. We use 2,165 samples for binary HC/MDD classification.

For each dataset, we randomly split 20% of the samples as the test set, while the remaining samples are used for retrieval database construction and model-related preparation. No supervised fine-tuning is performed on the LLM backbones.

D.2 Backbone LLMs and Baselines

We evaluate BrainAgent with both closed-source and open-source LLMs. For closed-source models, we use DeepSeek v3.2, Qwen3 Max, and Gemini 3.1 as the main backbones. For open-source models, we use Qwen3.5-35B, Qwen3.5-9B, and Gemma4-26B. In addition, we include several recent LLMs in the model generalization study, including GPT 5.3, Grok 4, Hunyuan 3, and Seed 2.0. For models without public technical papers, we report the official API/model identifiers used in our experiments, with access sources summarized in Appendix D.

For each backbone LLM, we compare BrainAgent with direct prompting. For closed-source models, we additionally compare with two common reasoning baselines:

- **CoT**[44]: Chain-of-Thought prompting, which encourages the model to produce step-by-step reasoning before prediction.
- **Reflection**[37]: A reflection-based baseline that asks the model to reconsider and revise its initial prediction after the first inference.

All methods use the same data split and input protocol. BrainAgent does not update the parameters of any LLM backbone and only augments inference through graph understanding, external retrieval, tool use, and reflection.

D.3 Evaluation Metrics

We evaluate all methods using Accuracy, Recall, Precision, and *pass@k*. Since all datasets in our experiments are binary classification tasks, we regard the disorder-related class as the positive class for ABIDE, ADHD, and Rest-meta-MDD, and use one class as the positive class for HCP following the same evaluation protocol. Let TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

Accuracy. Accuracy measures the overall proportion of correctly classified samples:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{5}$$

It reflects the general prediction performance of the model over all test samples.

Recall. Recall measures the proportion of positive samples that are correctly identified:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{6}$$

In disease-related brain network analysis, Recall reflects the model's ability to identify disorder-related subjects. However, high Recall alone does not necessarily indicate reliable prediction, since a model that predicts most samples as positive can also obtain high Recall.

Precision. Precision measures the proportion of truly positive samples among all samples predicted as positive:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{7}$$

Precision is particularly important in our setting because direct LLM inference often exhibits positive prediction bias. A low Precision with a high Recall indicates that the model tends to over-predict disorder-related labels and misclassify many healthy controls as positive.

pass@k. Following common evaluation protocols for LLM-based prediction, we also report *pass@k*. For each test sample, the LLM is queried k times under the same input setting. A sample is counted as correct under *pass@k* if at least one of the k generated predictions matches the ground-truth label:

$$\text{pass@}k = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[\exists j \in \{1, \ldots, k\}, \hat{y}_i^{(j)} = y_i\right], \tag{8}$$

where N is the number of test samples, y_i is the ground-truth label of sample i, and $\hat{y}_i^{(j)}$ denotes the j-th generated prediction. In our experiments, we report *pass@1* and *pass@3* when applicable.

For Accuracy under *pass@k*, we directly use the above criterion to determine whether each sample is correctly classified. For Recall and Precision under *pass@k*, we first aggregate the k generated predictions into the final sample-level prediction according to the same pass criterion and then compute Recall and Precision on the aggregated predictions. These metrics jointly evaluate overall prediction quality, sensitivity to positive samples, and the degree of over-prediction in LLM-based brain network analysis.

D.4 Detailed Analysis of Closed-Source LLM Results

This section provides a more detailed analysis of the closed-source LLM results on four public datasets, including ABIDE, ADHD, HCP, and Rest-meta-MDD. We use DeepSeek v3.2, Qwen3 Max, and Gemini 3.1 as backbone LLMs, and compare direct prompting, CoT, Reflection, and BrainAgent. The main results are reported in Table 2. Key findings and conclusive take-away messages are shown in Section K.

Direct LLM inference is still unreliable for brain network prediction. Across the four datasets, direct prompting often shows a typical positive prediction bias. In many cases, the model tends to classify most samples into the positive class, which leads to very high Recall but low Precision and low Accuracy. For example, on ADHD and HCP, direct prompting with DeepSeek and Qwen3 Max obtains nearly perfect Recall but poor Accuracy and Precision. This indicates that the LLM is not truly distinguishing disease-related and control samples, but instead tends to over-predict the positive label. Moreover, repeated inference does not always alleviate this issue, as the results of *pass@1* and *pass@3* are often identical for direct prompting. This suggests that simply sampling multiple outputs is insufficient when the model cannot effectively understand the underlying graph topology.

CoT provides limited improvement. Chain-of-Thought prompting encourages the LLM to produce intermediate reasoning before prediction, but its improvement is generally limited in this setting. On several datasets, CoT behaves similarly to direct prompting and still suffers from positive prediction bias. This is likely because the main challenge is not only whether the model can generate a reasoning chain, but whether the reasoning chain is grounded in reliable graph evidence and neuroscience knowledge. When the serialized brain graph is long and difficult to interpret, CoT may produce plausible but weakly grounded explanations, which do not necessarily improve the final prediction.

Reflection is more effective than CoT in many cases. Compared with CoT, Reflection usually brings larger gains. This shows that verification and revision are useful for LLM-based brain network analysis. Instead of directly trusting the first prediction, the reflection step asks the model to reconsider whether its conclusion is supported by the input evidence. This helps correct part of the initial reasoning errors and reduces unsupported predictions. However, Reflection alone still relies on the original serialized input and the model's internal knowledge, so its improvement is limited when the model lacks sufficient graph-topological understanding or domain-specific evidence.

BrainAgent reduces positive prediction bias and improves prediction reliability. BrainAgent consistently improves Accuracy and Precision across datasets and backbone models. Compared with direct prompting, BrainAgent no longer simply predicts most samples as positive. Instead, it uses topology-aware graph understanding, neuroscience knowledge retrieval, case retrieval, and reflection to produce more evidence-grounded predictions. The improvement is especially clear under *pass@3*, indicating that multi-round agentic interaction can make better use of repeated inference. Importantly, the decrease of Recall in some settings should not be interpreted as performance degradation. Since direct prompting often obtains high Recall by over-predicting the positive class, a moderate decrease in Recall together with higher Accuracy and Precision indicates a more balanced decision boundary.

BrainAgent remains a training-free framework with room for improvement. Although BrainAgent achieves clear improvements over direct prompting, CoT, and Reflection, its *pass@1* performance is still moderate on some datasets. This reflects the difficulty of directly applying general-purpose LLMs to specialized neuroimaging tasks without task-specific parameter updates. Unlike supervised brain graph learning methods, BrainAgent does not fine-tune the LLM or train a task-specific classifier. Its advantage lies in training-free transfer, flexible tool use, and interpretable reasoning. Future work may further improve this framework through domain-adaptive fine-tuning, better calibration, or reinforcement learning for agentic decision making.

E Detailed Interpretability Analysis

In this section, we provide a more detailed analysis of the interpretability evaluation. As shown in Figure 5, we compare the explanations generated by direct LLM inference and BrainAgent-enhanced inference across six backbone LLMs. The radar plots evaluate explanation quality from six

dimensions: **Domain Awareness**, **Completeness**, **Reflection Depth**, **Region Insight**, **Subgraph Insight**, and **Graph Insight**. These dimensions are designed to assess whether the generated explanation is not only correct in prediction, but also grounded, structured, and clinically meaningful.

Evaluation dimensions. **Domain Awareness** measures whether the explanation is grounded in the context of brain-network diagnosis, rather than only describing generic graph structures. For example, explanations that mention ASD/HC comparison, clinical relevance, functional connectivity, disease-related biomarkers, or neuroscience literature receive higher scores. **Completeness** evaluates whether the output contains a clear prediction, confidence score, sufficient supporting reasons, and a structured explanation. **Reflection Depth** measures whether the model explicitly checks its own analysis, prediction, and rationale. For BrainAgent, this score mainly comes from the reflection stage, while direct inference usually receives a low score because it does not include an explicit self-verification process. **Region Insight** evaluates whether the explanation discusses node-level or brain-region-level evidence, such as hubs, degree centrality, frontal regions, limbic regions, or parietal regions. **Subgraph Insight** measures whether the explanation analyzes community-level or functional-network-level evidence, such as DMN, FPN, modules, subgraphs, or within-network connectivity. **Graph Insight** evaluates whether the explanation captures global topological properties, such as small-worldness, global efficiency, path length, density, integration, and segregation. The final explainability score is computed as the average of these six dimensions.

Across the six evaluated LLMs, BrainAgent consistently obtains larger radar areas than direct inference. The improvement is especially clear in **Reflection Depth**, **Region Insight**, **Subgraph Insight**, and **Graph Insight**. This indicates that BrainAgent does not merely provide a label prediction, but decomposes the input brain network into multi-level evidence. In contrast, direct inference often relies on shallow descriptions, such as simply associating dense connectivity or preserved long-range connections with healthy controls. Such explanations may contain some domain-related words, but they usually lack detailed support from region-level hubs, subnetwork organization, and global graph topology.

Effect of multi-level graph understanding. The improvement in Region Insight, Subgraph Insight, and Graph Insight mainly comes from the topology-aware understanding stage. BrainAgent first converts the raw connectome into structured descriptions at three levels. At the region level, it identifies salient nodes, hub regions, and high-degree brain areas. At the subgraph level, it summarizes dense modules, community structures, and functional subnetworks. At the graph level, it describes whole-brain topological properties such as small-world organization, global efficiency, and integration–segregation balance. This multi-level decomposition makes the explanation easier to trace and reduces the risk that the LLM produces unsupported diagnostic claims.

Effect of reflection. The radar plots also show that BrainAgent substantially improves Reflection Depth. This is because BrainAgent explicitly performs a reflection step after the initial analysis. During reflection, the model checks whether its prediction and rationale are consistent with the extracted graph evidence, retrieved neuroscience knowledge, and similar cases. This mechanism helps the model detect overconfident or weakly supported conclusions. Direct inference lacks this verification stage, and therefore often produces one-step explanations without checking whether the reasoning is faithful to the actual graph structure.

Case study. We further analyze a representative ABIDE sample, denoted as ABIDE_2558, using Grok as the backbone model. The ground-truth label of this sample is ASD. Direct inference predicts the sample as HC, while BrainAgent correctly predicts it as ASD. The difference lies not only in the final prediction, but also in the quality of the reasoning process. Direct inference mainly reasons from a coarse observation that the graph appears dense and that long-range connectivity seems preserved, and therefore concludes that the sample is more similar to HC. This reasoning path is relatively shallow and does not sufficiently examine subnetwork organization or global topology.

By contrast, BrainAgent provides a more layered explanation. In the understanding stage, it first identifies high-connectivity nodes and hub-like regions. It then analyzes dense subgraphs and community-level structures, and finally relates these observations to graph-level properties such as small-worldness and global integration. In the final analysis, BrainAgent organizes high degree centrality, dense subgraphs, and altered global efficiency into an evidence chain supporting the ASD prediction. The reflection stage further checks whether this conclusion is consistent with the graph

evidence and retrieved knowledge. As a result, BrainAgent produces a more complete and verifiable reasoning chain.

Grok + BrainAgent


```
{
    "prediction": "ASD",
    "reason": [
        "The ABIDE dataset with fMRI modality and AAL template shows high degree centrality in nodes
             like 0, 16, and 28, which are associated with frontal and limbic regions, indicating
             hyperconnectivity patterns commonly linked to ASD rather than HC.",
        "Dense sub-graphs in regions related to executive functions (e.g., nodes 10-19) and emotional
             processing (e.g., nodes 72-81) suggest altered local clustering and community structures,
             aligning with known ASD biomarkers as confirmed in the previous analysis and retrieved
             knowledge on sensorimotor hyperconnectivity.",
        "The overall graph exhibits small-world properties with high connectivity and potential changes
             in global efficiency, such as increased average path length, which are frequently observed
              in ASD networks compared to HC, as summarized in the report.",
        "The previous report emphasizes reliance on these confirmed patterns for ASD prediction,
             including increased connectivity and sub-graph densities, without contradictory evidence
             from retrieve_case, supporting a diagnosis of ASD based on the task of ASD diagnosis."
    ],
    "confidence": "5"
}
```


Direct Grok Inference


```
{
    "prediction": "HC",
    "reason": [
        "The ABIDE dataset with fMRI modality and AAL template shows a brain network with a high number
             of connections (e.g., numerous edges like (0,1,1.000) and (0,10,1.000)), indicating dense
             functional connectivity, which is often associated with healthy controls in ASD diagnosis
             tasks.",
        "Statistical analysis of the graph reveals an extensive degree distribution, with nodes such as
             0 connected to multiple regions (e.g., 1, 10, 11), suggesting preserved long-range
             connectivity that is typically more disrupted in ASD subjects.",
        "The graph's overall structure, with no evident isolated nodes and a balanced distribution of
             edges across regions, aligns with patterns observed in HC groups, as opposed to the
             hypoconnectivity in social and cognitive networks commonly seen in ASD."
    ],
    "confidence": "3"
}
```


These results suggest that BrainAgent improves interpretability in two complementary ways. First, it transforms the raw brain graph into hierarchical evidence, including region-level, subgraph-level, and graph-level information. Second, it adds a reflection stage to verify whether the final prediction is supported by the extracted evidence. Therefore, BrainAgent's explanations are more structured, more traceable, and closer to the type of evidence-driven reasoning required in brain network analysis.

F Details of Hierarchical Augmented Retrieval for Neural Knowledge (HARK)

We design two distinct function calls aiming to minimize knowledge gap between LLM's general domain and certain task domain. HARK injects domain knowledge via general Retrieval Augmented Generation (RAG), narrowing certain domain gap between vanilla LLMs and specialized brain network analysis.

F.1 Knowledge Base Construction

Knowledge base construction: HARK is a novel RAG module tailored for brain network analysis. Our knowledge base consists of a set of atomic factual meta knowledge points in neuroscience. Formally, each meta knowledge data consists of knowledge content and the involved brain regions. We first collect relevant academic papers and professional medical manuals (e.g. DSM5) as source documents according to different tasks (e.g. MDD diagnosis, ADHD diagnosis). Then we prompt

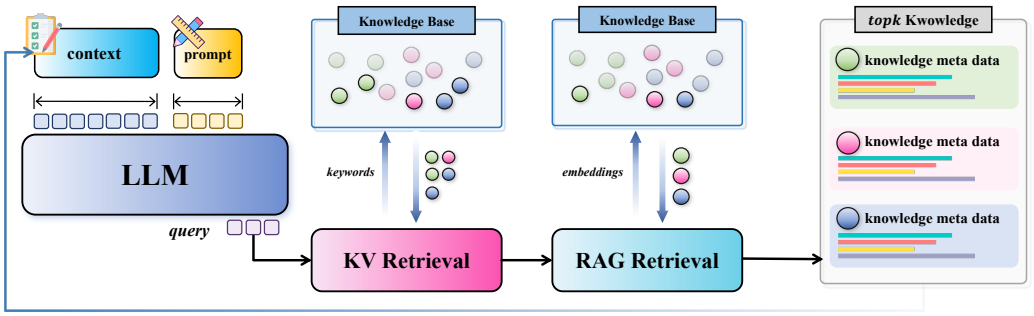


Figure 7: Process of HARK. HARK is a two-stage, hierarchical augmented retrieval for neural knowledge. In the first stage, IoU scores of concerned brain region sets between query and key knowledge is calculated for coarse filtering. Then in the second stage, similarity from embedding level is calculated for fine-grained recall.

LLM to extract useful knowledge points from source documents. The core logic of our prompt follows [32]'s practice. We employ LLMs to extract and decompose knowledge points from source data, until each decomposed unit only contains core neuroscience knowledge and the corresponding involved brain regions. Here we conduct document collection and knowledge extraction according to different task types, thus the resulting knowledge set is further categorized by task type. In addition, considering that common knowledge points exist across different brain disease tasks, we also construct a general knowledge set for universal brain network analysis (denoted as "general").

Knowledge base refinement: After the construction of knowledge sets, we adopt different LLM to score, filter, and refine current knowledge data from four dimensions: accuracy, usefulness, linguistic redundancy, and professionalism. For knowledge extraction, we use Deepseek v3.2, and for knowledge refinement, we utilize Kimi k2 with thinking mode. Furthermore, we map each brain region to its corresponding ID in the AAL template and unify different naming aliases of the same brain region which may cause bias in unprocessed knowledge data. Finally, the obtained knowledge points are further checked manually. In this case, we construct a knowledge base consisting of 2000 high-quality knowledge points in total, with average length of each knowledge point 53.2 words. Note that in this work the scale of whole knowledge base is limited, and as we focus on complete workflow, improvement of the dataset's quality and scale can be explored in future works.

F.2 Hierarchical Augmented Retrieval

We perform knowledge retrieval on the constructed knowledge base mentioned above. In summary, as is shown in Figure 7 the retrieval framework of HARK integrates semantic-based and representation-based search, realizing a coarse-to-fine two-stage retrieval strategy. The first stage is semantic-based retrieval, whose core motivation is to filter relevant knowledge points from the corresponding knowledge base for given specific task. We compute the similarity between the brain region set involved in the LLM query (R_q) and sets associated with each knowledge point (R_k). Then a threshold is adopted for filtering, where one knowledge point will be recalled in this stage if the similarity exceeds the predefined threshold value. We adopt the IoU score as the similarity metric and the threshold is set 0.5. Finally, this stage returns the indices of all recalled knowledge points.

The second stage follows the conventional RAG pipeline. We employ a text representation model to compute the similarity between the query and knowledge points from embedding level, where only the knowledge points recalled in the previous stage are used for similarity calculation. We adopt PubMedBERT [13] for text representation, and finally return the top-k knowledge points ranked by similarity. Here we set top-k value as 5. For knowledge from task-specific knowledge base, we follow this two-stage retrieval process. Besides, for knowledge from general knowledge base, we directly retrieve top-k knowledge points from embedding level. In other words, BrainAgent will receive $2 \times$ top-k knowledge points in total from HARK module.

G Details of Graph Foundation Model for Brain Network Data (GFM)

In this section, we provide details of the pretraining process for the Graph Foundation Model (GFM).

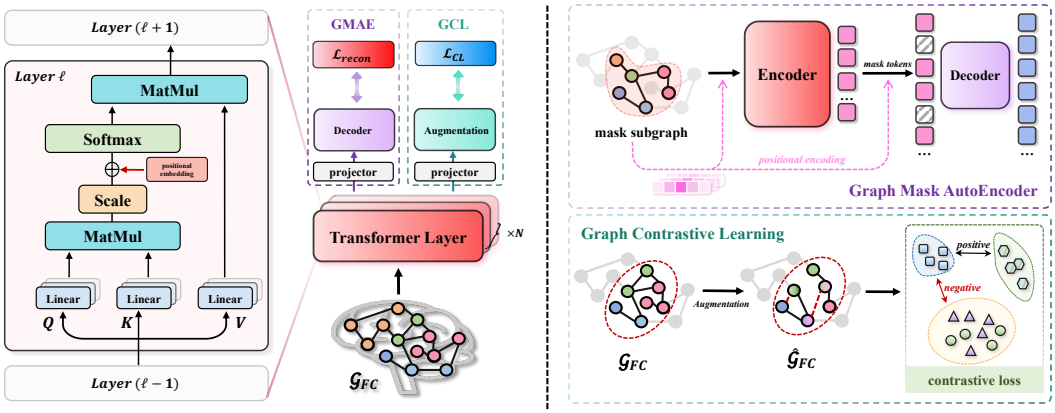


Figure 8: (Left) Structure of Graph Foundation Model. (Right) Training objective of GFM.

G.1 Overall Pretraining Objective

The pretraining process consists of two joint optimization objectives: Graph Mask Autoencoder (GMAE) and Graph Contrastive Learning (GCL).

$$\mathcal{L}_{pretrain} = \mathcal{L}_{recon} + \mathcal{L}_{GCL} \tag{9}$$

G.1.1 Graph Masked Autoencoder (GMAE)

The main idea of Graph Masked Autoencoder is to reconstruct features of the whole graph from its masked subgraph. We follow the practice from standard MAE [15], by adding random masks on input graph, including node mask and edge mask [17].

Formally, for an input graph $\mathcal{G}$ with N nodes, we denote it as $\mathcal{G} = (\mathbf{X}, \mathbf{A}, \mathcal{V}, \mathcal{E})$. For nodes, we randomly select a subset as masked nodes: $\mathcal{V}_M \in \mathcal{V}$. Nodes in $\mathcal{V}_\mathcal{M}$ are replaced with learnable vector $\mathbf{x}_{[M]} \in \mathbb{R}^d$, which is initialized randomly. The node features is then defined as:

$$\tilde{\mathbf{x}}_i = \begin{cases} \mathbf{x}_{[M]}, & v_i \in \mathcal{V}_M \\ \mathbf{x}_i, & \text{otherwise} \end{cases} \tag{10}$$

For edges, we apply masking by dropping edges randomly. Each edge is dropped independently with probability p. Finally we get masked subgraph $\tilde{\mathcal{G}} = (\tilde{\mathbf{X}}, \tilde{\mathbf{A}})$ with corrupted feature and structure. We add positional embeddings (denoted as **PE**) on $\tilde{\mathbf{X}}$ which is preprocessed on the whole graph.

The embeddings of un-masked nodes are obtained via encoder (GFM):

$$\mathbf{H}_{encoder} = \text{GFM}\left(\tilde{\mathcal{G}}, \mathbf{PE}(\mathcal{G})\right) \tag{11}$$

For implementation of GFM, we employ N-layer Graph Transformer as our backbone, with each layer composed of multi-head attention and a two-layer feed forward network (FFN).

The input of the decoder is tokens from the whole graph consisting of (a) encoded embeddings for unmasked nodes and (b) masked node tokens. For masked node tokens, they share a uniform learnable vector. Moreover, we add positional embeddings for these masked nodes. For implementation of decoder, we select M-layer Transformer for reconstruction, where $N > M$.

The reconstruction target is to predict the raw features of masked nodes, and the loss function is the mean squared error (MSE) between the decoded outputs and raw features:

$$\mathcal{L}_{recon} = \frac{1}{|\mathcal{V}_M|} \sum_{i=1}^{\mathcal{V}_M} \left|\left| \tilde{\mathbf{x}}_i - \mathbf{x}_i \right|\right|^2 \tag{12}$$

G.1.2 Graph Constrastive Learning (GCL)

Along with reconstruction task, we adopt contrastive learning method to enhance model's capability under perturbations. Here we directly adopt INFONCE as GCL method. Formally, we exert perturbations from two different augmented views by randomly dropping out features and edges, and generate augmented graph $\mathcal{G}' = (\mathbf{X}', \mathbf{A}')$. Then both raw graph $\mathcal{G}_i$ and augmented graph $\mathcal{G}_i'$ are sent into shared GFM encoder and get graph-level embeddings denoted as $z_i, z_i' \in \mathbb{R}^d$. The objective of GCL is to maximize the similarity between embeddings from same graph while keeping the distance from other samples in one Batch B:

$$\mathcal{L}_{GCL} = -\frac{1}{B} \sum_{i=1}^{B} \log \frac{\exp\left(\frac{\text{sim}(z_i, z_i')}{\tau}\right)}{\exp\left(\frac{\text{sim}(z_i, z_i')}{\tau}\right) + \sum_{k \neq i} \exp\left(\frac{\text{sim}(z_i, z_k)}{\tau}\right)} \tag{13}$$

Parameters	**Values**
Number of input nodes	90
Number of graphs	4760
Number of GFM layers (N)	6
Positional encoding (**PE**)	`RWSE`
Hidden dim (d)	128
Feed forward dim	256
Number of attention heads	8
Normalization	`LayerNorm()`
Dropout	0.3
Number of GMAE decoder layers (M)	2
Epochs for pretraining	500
Batch size (B)	128
Mask ratio for GMAE	0.7
Temperature for GCL (τ)	1.0

Table 5: Parameters for GFM and training process.

G.2 Details on Experimental Settings

The backbone of GFM follows the architecture of Graph Transformer, with its encoder constructed by stacking N Transformer blocks (Here, $N = 6$). For Graph Transformers, different positional encoding (PE) strategies are often utilized to inject topological information for graphs. Here, we use Random Walk Structural Encoding (RWSE) for implementation, following BrainGFM's practice [45]. As we use AAL template for preprocessing, each input brain graph maintains exactly 90 brain regions (nodes). The layer of GFM is 6 and the hidden dim is 128. For each Transformer block, the number of attention heads in each multi-head self-attention layer is 8. More parameter settings are listed in Table 5.

Total epochs of pretraining process is 500, and here the pretraining loss is the sum from GMAE and GCL with no hyper-parameters for adjusting their weights. The figure of training loss curve is shown in Figure 9.

H Details of Case Augmented Retrieval via Dual-Modality Recall (CARD)

Corresponding to HARK, we design CARD, aiming to achieve knowledge enhancement from relevant samples (cases). Prior works such as RAGraph [20] and RAG4GFM [43] combine graph pre-training

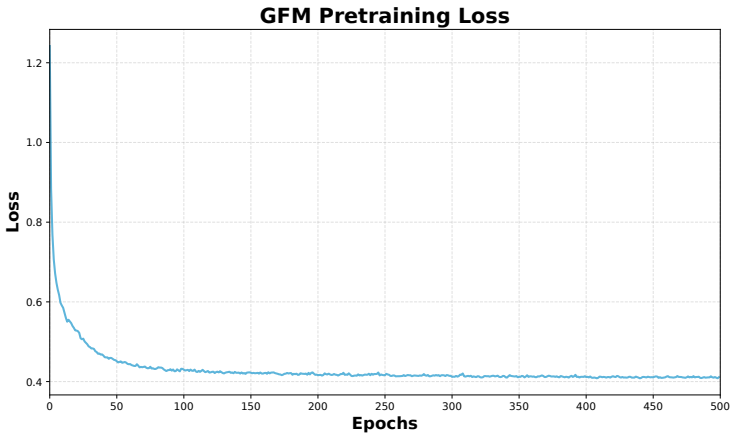


Figure 9: Visualization of loss curve on GFM.

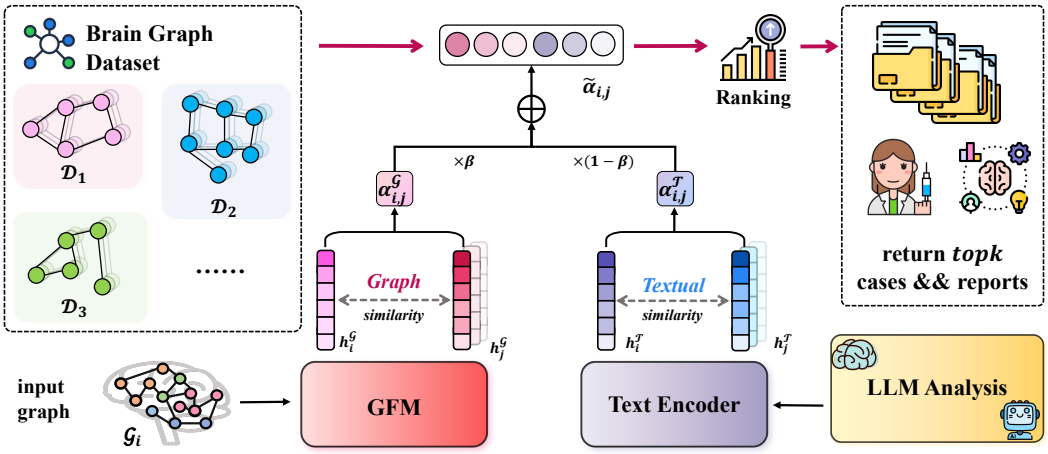


Figure 10: Process of CARD. CARD is a case-level augmented retrieval framwork via dual-modality recall. Graph-level (from GFM) and text-level (from LM) similarities are calculated parallel and the final ranking score is the mixing of two values.

with retrieval augmentation. Given an input graph, these methods retrieve similar graphs from a constructed memory to improve downstream graph reasoning. CARD follows this general idea but performs retrieval at the subject level for brain network analysis.

Encouraged by this, core innovation of CARD is to compute similarity and retrieve most relevant cases at the sample level. For a given brain network, CARD calculates the similarity between the input and samples in the corresponding dataset, then retrieves the most similar sample data. This further improves the accuracy and interpretability of the subsequent analysis and reasoning of BrainAgent.

As is shown in Figure 10, the overall framework of CARD follows a dual-modality retrieval process. Inspired by RecGPT [53], we calculate embedding similarity from a mixing of both graph-level and text-level, leading to finer-grained feature matching.

H.1 Brain Graph Dataset Construction

The first step of CARD is to construct a brain graph dataset as case memory based on raw graph format data $\mathcal{G}$. We first split the dataset into training and test sets, following the partitioning setting described in Section 3.1. Then for train dataset (also denoted as support dataset), we construct high-quality text-modality report as meta data for case memory. Similar to HARK's construction pipeline, we first generate report for each data then refine them before final manual check.

Dataset construction: For content generation, we prompt LLM to generate description report for given brain network data. Brain networks are organized in a textual form similar to Understanding part, and we also adopt the analytical tools introduced before for augmentation. The logic of prompt design is roughly consistent with that in Understanding part 2.2, which requires LLM to analyze brain network graphs and summarize the results into a report. The difference is that we additionally feed the corresponding labels into the LLM. We attribute this to that generating analytical reports based on given labels achieves lower difficulty and higher accuracy than direct analysis [37] as it can be regarded as finding proofs instead of pure reasoning. Since this process does not involve test dataset, there is no risk of data leakage. For report generation we utilize Deepseek v3.2 as LLM. The output format of report strict follows json format which contains information from region level, subgraph level, graph level, case level and a final conclusion.

Dataset refinement: After the LLM generation, we prompt another LLM to check and refine. Refinement process considers the following dimensions: occurrence of factual errors, correctness of the generated analysis report, whether the generated content is redundant and useless, and professionalism of the language style. Here, we use Kimi k2 with thinking mode as LLM. Moreover, we prompt LLM to output a tag `if_manual` besides from refined content, indicating if the results needs further manual check and correction. Those tagged "True" will be checked and modified via human experts in the final stage.

H.2 Case Augmented Retrieval via Dual-Modality Recall

Overall architecture of CARD is shown in Figure 10, with the whole pipeline consisting of two parallel modal pathways: graph data and text data. More details are discussed below:

Dual-modality embedding: The input of CARD contains raw data $\mathcal{G}_i$ and query from BrainAgent $\mathcal{Q}_i$. For graph data, we utilize pretrained GFM to get graph-level embeddings $h_i^{\mathcal{G}}$ and for text query we utilize PubmedBERT for textual embeddings $h_i^{\mathcal{T}}$, which is shown in Equation 14:

$$\begin{cases} h_i^{\mathcal{G}} = \text{Readout}\left(\text{GFM}(\mathcal{G}_i)\right) \in \mathbb{R}^{d_G} \\ h_i^{\mathcal{T}} = \text{PubmedBERT}\left(\mathcal{Q}_i\right) \in \mathbb{R}^{d_T} \end{cases}, \tag{14}$$

where Readout$(\cdot)$ denotes pooling function. Here $d_g = 128$ and $d_T = 768$. It is worth noting that the representations of the test dataset are computed in real time ($\mathcal{G}_i$ and $\mathcal{Q}_i$). On the other hand, for training dataset, the corresponding textual data are generated reports. Therefore, we can pre-compute and store these representations locally, thereby further reducing resource consumption during inference phase. For j-th data in training dataset, the graph-level embedding is denoted as $h_j^{\mathcal{G}}$ and text-level embedding $h_j^{\mathcal{T}}$.

Mixing retrieval for graph-text similarity: Our retrieval strategy is the mixing of graph and text retrieval process. For given i-th data, we calculate cosine similarity for each data j in training dataset. Notice that we partition the data by different tasks at the initial stage. In other words, the bran graph dataset is organized according to task categories, and the retrieval and recall process only needs to be performed within certain dataset. Similarity from graph-level and text-level are denoted as $\alpha_{i,j}^{\mathcal{T}}$ and $\alpha_{i,j}^{\mathcal{T}}$ respectively. The final mixing retrieval score is defined as:

$$\tilde{\alpha}_{i,j} = \beta \cdot \alpha_{i,j}^{\mathcal{G}} + (1-\beta) \cdot \alpha_{i,j}^{\mathcal{T}}, \tag{15}$$

where β controls the balance between graph-level and text-level matching. In our implementation, β is fixed to 0.5. Finally, we retrieve the top-k samples according to the computed similarity $\tilde{\alpha}_{i,j}$ and return their corresponding reports. For CARD, we test results on different top-k ranging in $\{3, 4, 5, 6\}$ and found that best results for top-k is 3, indicating that if the retrieval process is designed useful, only a few reference answers are enough.

I Prompt Template

In this section we provide our designed prompt template for BrainAgent in different parts.

1. **Understanding**: Prompt template for Understanding part is shown in Prompt I. The prompt consists of basic data information and statistical results from defined analysis function calls.

Output is required as strict json format with region-level, subgraph-level and graph-level analysis.

2. **Think and Report**: Prompt template for Think part is shown in Prompt I. BrainAgent is required to perform reasoning before action. Meanwhile, it should also maintain and update a fixed-size report which stores useful history information.
3. **Action**: Prompt template for Action part is shown in Prompt I. LLM is prompted to select corresponding function call from available function tools. Meanwhile, concrete queries are required as parameters for different function call. `retrieve_knowledge` stands for HARK module and `retrieve_case` stands for CARD module.
4. **Analysis**: Prompt template for Analysis part is shown in Prompt I. We prompt LLM to make prediction and give corresponding reasons.
5. **reflection**: Prompt template for Reflection part is shown in Prompt I. Reflection is made after final analysis for correctness and refinement. LLM is asked to reflect on different generation contents, including LLM analysis, prediction and reasons in "Analysis" part.

Prompt A: Prompt for Understanding

 # Role Definition

You are an expert in neuroscience and machine learning. You can distinguish and analyze difference for given brain network data which takes form of graph.

 # Task Description

Now give you brain network data which forms as graph, you need to analyze features of the network related to the given data modality.

 # Input Data

- **data modality**:
{data_modality}
- **task**:
{task}
- **template**: AAL template
- **label**: {label_1}, {label_2}
- **data format description**:
{data_description}
- **data**:
{graph_info}
- **statistical results**: To further improve accuracy of this part and lower hallunation, some statistical results are provided below. You can directly use them to guide your analysis **INSTEAD OF CALUCULATING THEM AGAIN ON YOUR OWN.**
{statistical_results}

 # Requirements

Your analyze should contain three dimensions: region-level (node), sub-graph-level and graph-level.

- **region-level**: Analyze the features of each region (node) in the brain network, such as degree, centrality, clustering coefficient, etc. Identify key regions that may play important roles in the network.
- **sub-graph-level**: Identify and analyze important sub-graphs or communities within the brain network. Discuss their potential functions and interactions.
- **graph-level**: Analyze the overall structure and properties of the brain network.

 # Output Requirements

Your output must strictly follow json format below:

```
```json
{{
"region": ["region feature 1", "region feature 2", ...],
"sub-graph": ["subgraph feature 1", "subgraph feature 2", ...],
"graph": ["graph feature 1", "graph feature 2", ...]
}}
```
```

Now start your analysis.

Prompt B: Prompt for "Think" and "Report"

 # Role Definition

You are an expert in neuroscience and machine learning. You can distinguish and analyze difference for given brain network data which takes form of graph.

 # Task Description

Now give you brain network data, you need to give the correct prediction and give your reasons as best as you can. For those knowledge you are not familiar with, you can use tools to support your answer.

 # Input Data

{data}

 # Last Action and Following Observation

report:
{report}
action:
{action}
observation:
{observation}

 # Requirements

Your main solution should strictly follow 'think' -> 'report' -> 'action' loop.
- think: Think is reasoning about the current situation. Your thinking process should combine 'input data', 'observation' from latest 'action' and 'report'.
- report: After thinking, you should maintain a report. It is the summarization of the current information and should include previous content of report and your current thinking. If the word length is more than 300 words, you should summarize and only extract most valuable key information.
Format of report can follow contents below which MUST BE a **markdown style string**:

```
1. Analyze: Your analysis.
2. Action 1: Summarization of action 1 and observation.
...
```

- action: Your available action must be within these tools below:
{tools}

 # Output Requirements

Your output must strictly follow json format below. **DO NOT MODIFY ITS STRUCTURE:**

```json
{{
"think": "your thinking process, within 100 words",
"report": "your report, within 300 words",
}}
```

Now your step is think.

Prompt C: Prompt for "Action"

 # Role Definition

You are an expert in neuroscience and machine learning. You can distinguish and analyze difference for given brain graph network data.

 # Task Description

Now give you brain network data, you need to give the correct prediction and give your reasons as best as you can. For those knowledge you are not familiar with, use tools to support your answer before giving your final answer.

 # Input Data

data:
{data}
think:
{think}
report:
{report}

 # Requirements

Your main solution should strictly follow 'think' -> 'report' -> 'action' loop.
- think: Think is reasoning about the current situation. Your thinking process should combine 'input data', 'think' content and 'report'.
- report: After thinking, you should maintain a report. It is the summarization of the current information and should include previous content of report and your current thinking.
- action: Your available action must be within these types below:

 # current tools

`<function_call></function_call>` denotes name of the function. NOTE that tool name after it is the name of the tool, DO NOT DIRECTLY OUTPUT function_call.
`<args></args>` denotes arguments of the function.
`<description></description>` denotes detailed description and usage of the tool.

{tools}

 # Note

- For selection of tools, you should combine information from both 'data', 'think' and 'report' to get more comprehensive information before making decision.
- You should select '`analysis_final`' only after '`retrieve_knowledge`' and '`retrieve_case`'.

 # Output Requirements

Your output must strictly follow json format below:

```json
{{
"action": "action to take",
"args": {{
"arg1": "val1",
"arg2": "val2",
...
}}
}}
```

Now your step is action.

Prompt D: Prompt for "Analysis"

 # Role Definition

You are an expert in neuroscience and machine learning. You can distinguish and analyze difference for given brain network data which takes form of graph.

 # Task Description

Now give you brain network data which forms as graph, you need to analyze features of the network related to the given data modality before finally making a prediction.

 # Input Data

- dataset:
{dataset}
- data_modality: fMRI
- template: AAL
- task:
{task}
- prediction label:
{label_1}, {label_2}
- report:
{report}

 # Requirements

Your analyze should consider complete data and previous analysis, including:
(1) basic information including 'dataset', 'data_modality', 'template', 'task', 'prediction label';
(2) 'data' which contains statistical analysis of the graph;
(3) 'report' which contains summarization of previous analysis;

 # Output Requirements

Your output should strictly follow json format below:
```json
{{
"prediction": "precition of the data, should be strictly aligned to 'prediction label'",
"reason": ["reason 1", "reason 2", ...],
"confidence": "a number between 1-5 indicating confidence for your judgement. 1 for low confidence, 5 for high and 3 for medium."
}}
```

Now start your analysis.

Prompt E: Prompt for "Reflection"

 # Role Definition

You are an expert in neuroscience and machine learning. You can distinguish and analyze difference for given brain network data which takes form of graph. Besides, you are an advanced reasoning agent that can improve based on self refection.

 # Task Description

Now give you brain network data which forms as graph, as well as its analysis report from LLM. You need to reflect on the generation from LLM based on the input data and other neuroscience knowledge and make refinement and correction if necessary.

 # Raw Data info
{data_info}

 # LLM analysis
{llm_analysis}

 # LLM reasoning and prediction
{llm_output}

 # Note

- LLM output include 'llm analysis' 'prediction', 'reason' and 'confidence'.
- Requirements on your reflection:
- LLM' generation can be reflected in at least these dimension:
1. If the analysis from graph data is correct.
2. If the reason is valid based on the data and corresponding analysis.
3. If the final prediction is correct.
- Confidence from LLM generation is also a factor indicating llm's own certainty for its prediction.
- To avoid hallucinations, you need to combine your judgement with official neuro-knowledge. You can also refer to external knowledge, tools and resources to improve your reflection.
- After reflection, you should also finally make your own prediction for refinement and correction.
REMEMBER that your judgement of the data is based on your own analysis and prediction, as well as refined and reasoning process.

 # Output Requirements

Your output MUST strictly follow json format below:
```json
{{
"thinking": "Key logic of your detailed thinking process",
"judgement of prediction": "Your judgement of the prediction from LLM, a score beween 0 and 1",
"judgement of reason": "Your judgement of the reason from LLM, a score beween 0 and 1",
"reasons": ["Reasons for your reasons on the judgements"],
"prediction_raw": "The original prediction from LLM",
"prediction_refined": "Your refined prediction which value can only between ['label1', 'label2']"
}}

```
Now start your reflection.

J Visualization of Reasoning Results

To better illustrate how BrainAgent performs interpretable brain-network reasoning, we visualize the intermediate reasoning trajectory of one representative sample from the ABIDE dataset. In this case, we use **Gemini 3.1** as the backbone LLM. The ground-truth label of this sample is HC, while direct LLM inference predicts it as ASD. In contrast, BrainAgent decomposes the input brain network into multi-level structural evidence, including region-level, subgraph-level, and graph-level descriptions, which provides a more traceable reasoning process for the final prediction.

Understanding-stage Reasoning Result


```
{
    "region": [
        "High-degree hubs: Node 32 (degree 15), Node 56 (degree 15), Node 57 (degree 16), Node 58 (
             degree 17), Node 59 (degree 14), Node 61 (degree 14), Node 62 (degree 14), Node 63 (degree
              12), Node 84 (degree 14), Node 85 (degree 14) form a densely connected core in posterior
             brain regions",
        "Frontal lobe nodes (0-23) show moderate connectivity with average degree 5.2, with strongest
             connections between nodes 10, 12, 13, 18, 19, 60-65",
        "Occipital and parietal nodes (42-55) form a tightly interconnected cluster with average degree
              6.8, particularly nodes 42-49 showing near-complete connectivity",
        "Temporal lobe nodes (80-89) show moderate connectivity with average degree 4.3, with nodes 84,
              85, 88, 89 serving as connection hubs",
        "Subcortical nodes (70-79) form a distinct module with average degree 5.6, centered around
             nodes 75, 76, 77",
        "Node 32 (right hippocampus) shows extensive connections to frontal (0,1,10,12,18,19,60-64),
             temporal (56-69,84,85), and parietal regions",
        "Node 75 (right thalamus) serves as a major subcortical hub with 14 connections to frontal,
             parietal, and other subcortical regions"
    ],
    "sub-graph": [
        "Frontal network (nodes 0-23): Density 0.42, strongly interconnected with particular strength
             in prefrontal regions (nodes 10-19)",
        "Default Mode Network components (nodes 31-33, 56-69, 80-81, 84-89): Intra-density 0.395,
             showing moderate internal connectivity",
        "Occipital-parietal cluster (nodes 42-55): Near-complete subgraph with density 0.92,
             representing highly synchronized visual processing regions",
        "Subcortical module (nodes 70-79): Density 0.56, centered around thalamic and basal ganglia
             regions with strong internal connections",
        "Fronto-parietal network connections: DMN-FPN inter-density 0.106, indicating moderate cross-
             network integration",
        "Temporal lobe cluster (nodes 80-89): Density 0.38, with stronger posterior temporal
             connections than anterior",
        "Cingulo-opercular network components visible in nodes 24-31 with connections to both frontal
             and subcortical regions"
    ],
    "graph": [
        "Small-world architecture: ($\sigma$ = 3.0395 indicates strong small-world properties with high
              clustering and efficient path lengths",
        "High modularity: Q = 0.4696 shows clear community structure with distinct functional modules",
        "Efficient global integration: Global efficiency = 0.5224 suggests good information transfer
             capability",
        "High local clustering: Average clustering coefficient = 0.5943 indicates strong local
             connectivity within modules",
        "Characteristic path length = 808988.9551 suggests some long-distance connections but efficient
              local processing",
        "Network shows typical ASD-related features: preserved small-worldness but altered modular
             organization compared to HC",
        "DMN integrity: Moderate intra-density (0.395) with reduced long-range connections
             characteristic of ASD",
        "Fronto-subcortical connectivity: Strong connections between prefrontal regions and thalamus/
             basal ganglia"
    ]
}
```


This example shows that BrainAgent does not directly rely on a single-step label prediction. Instead, it first converts the raw connectome into structured multi-level evidence. The region-level analysis highlights salient brain regions and hub nodes, the subgraph-level analysis summarizes local community organization, and the graph-level analysis captures whole-brain topological properties. These intermediate outputs make the reasoning process more transparent and provide interpretable evidence for subsequent retrieval, reflection, and final analysis.

J.1 Visualization of the First-round Reasoning

After the understanding stage, BrainAgent first summarizes the current structural evidence into a compact report. Based on this report, the agent decides to call the `retrieve_knowledge` function to obtain task-related neuroscience knowledge. The retrieved knowledge is returned as the `observation` field and is used as external evidence for the next reasoning round.

Report-stage Reasoning Result (Round 1)


```
{
    "think": "The input data describes a brain network with strong small-world properties ($\sigma$
          =3.0395), high modularity (Q=0.4696), and efficient global integration. Key features include:
          a densely connected posterior core (nodes 32,56-59,61-63,84,85), strong occipital-parietal
          clustering (density 0.92), and preserved but altered DMN connectivity. The report mentions '
          typical ASD-related features' including preserved small-worldness but altered modular
          organization and reduced DMN long-range connections. I need to retrieve knowledge about ASD
          brain network patterns and compare with similar cases.",
    "report": "## Brain Network Analysis Report\n\n1. **Analyze**: Initial assessment reveals a brain
          network with:\n - Strong small-world architecture (($\sigma$=3.0395) indicating efficient
          information processing\n - High modularity (Q=0.4696) with distinct functional communities\n -
           Densely connected posterior hub regions (nodes 32,56-59,61-63,84,85)\n - Near-complete
          occipital-parietal cluster (density 0.92)\n - Moderate DMN connectivity (intra-density 0.395)
          with reduced long-range connections\n - Fronto-subcortical integration via thalamic hub (node
          75)\n - Data suggests 'typical ASD-related features' with preserved small-worldness but
          altered modular organization"
}
```


Action-stage Reasoning Result (Round 1)


```
{
    "action": "retrieve_knowledge",
    "args": {
        "regions_q": ["32", "56", "57", "58", "59", "61", "62", "63", "84", "85", "75"],
        "query": "Brain network shows strong small-world properties (($\sigma$=3.0395), high modularity (Q=0.4696), and
              efficient global integration. Key features include a densely connected posterior core (high-degree hubs in
              hippocampus, temporal, and parietal regions), near-complete occipital-parietal cluster (density 0.92),
              moderate DMN intra-connectivity (0.395) with reduced long-range connections, and strong fronto-subcortical
              integration via thalamic hub. The report mentions 'typical ASD-related features' with preserved small-
              worldness but altered modular organization. Need to retrieve knowledge on ASD brain network patterns: how
              small-worldness, modularity, DMN connectivity, and hub organization differ in ASD compared to healthy
              controls."
    },
    "observation": [
        "ASD is primarily characterized by repetitive behaviors, with overconnectivity in motor cortex and
              temporoparietal junction.",
        "ASD is primarily characterized by sensory deficits, with increased connectivity between thalamus and sensory-
              motor networks.",
        "ASD is primarily characterized by emotional deficits, with underconnectivity between amygdala and inferior
              parietal lobule.",
        "ASD is primarily characterized by sensory-motor deficits, with underconnectivity between dorsal attention and
              somatomotor networks.",
        "Default mode network anticorrelated with frontoparietal network in resting-state, consensus from meta-studies.",
        "Inferior frontal network is primarily characterized by language processing, with high connectivity among IFG
              parts in healthy brains.",
        "Inferior frontal network is primarily characterized by language processing, with high connectivity among IFG
              parts in healthy brains.",
        "Frontoparietal Network (FPN) is primarily characterized by executive control, with the most robust and
              reproducible resting-state brain network biomarker being connectivity between middle frontal gyrus and
              inferior frontal operculum, which is significantly correlated with inhibitory control.",
        "Frontoparietal Network (FPN) is primarily characterized by executive control, with the most robust and
              reproducible resting-state brain network biomarker being connectivity between middle frontal gyrus and
              superior parietal lobule, which is significantly correlated with cognitive flexibility."
    ]
}
```


J.2 Visualization of the Second-round Reasoning

The second-round reasoning result is shown below. After receiving neuroscience knowledge from the first round, BrainAgent updates the `report` field through reflection and further calls the `retrieve_case` function. This step retrieves top-k similar samples from the ABIDE case memory (here top-k=5), together with their corresponding reports, so that the agent can compare the current brain network with task-specific precedents.

Report-stage Reasoning Result (Round 2)


```
{
   "think": "The input data shows a brain network with high modularity, small-world properties, and specific hub
         patterns. The report mentions 'typical ASD-related features' with preserved small-worldness but altered
         modular organization. The retrieved knowledge includes ASD characteristics like sensory-motor deficits and DMN-
         FPN anticorrelation. I need to analyze if this network aligns with ASD patterns, focusing on connectivity
         alterations in DMN, FPN, and subcortical regions.",
   "report": "## Brain Network Analysis Report\n\n1. **Analyze**: Initial assessment reveals a brain network with:\n -
         Strong small-world architecture ($\sigma$=3.0395) indicating efficient information processing\n - High
         modularity (Q=0.4696) with distinct functional communities\n - Densely connected posterior hub regions (nodes
         32,56-59,61-63,84,85)\n - Near-complete occipital-parietal cluster (density 0.92)\n - Moderate DMN
         connectivity (intra-density 0.395) with reduced long-range connections\n - Fronto-subcortical integration via
         thalamic hub (node 75)\n - Data suggests 'typical ASD-related features' with preserved small-worldness but
         altered modular organization\n\n2. **Action 1**: Retrieved knowledge on ASD and network biomarkers.
         Observations indicate ASD involves sensory-motor deficits, DMN-FPN anticorrelation, and executive control
         linked to FPN connectivity. Key points: ASD shows overconnectivity in motor/temporoparietal areas,
         underconnectivity in dorsal attention-somatomotor networks, and DMN anticorrelated with FPN."
}
```


Action-stage Reasoning Result (Round 2)


```
{
  "action": "retrieve_case",
  "args": {
    "query_t": "Brain network shows high modularity (Q=0.4696), strong small-worldness (sigma=3.0395), and preserved
         local clustering (0.5943) but altered modular organization. Key features include densely connected posterior
         hubs (nodes 32, 56-59, 61-63, 84, 85), a near-complete occipital-parietal cluster (density 0.92), moderate DMN
          intra-density (0.395) with reduced long-range connections, and strong fronto-subcortical integration via
         thalamic hub (node 75). These patterns match reported ASD characteristics: preserved small-world architecture
         but altered community structure, DMN underconnectivity, sensory/visual overconnectivity, and subcortical hub
         prominence."
  },
  "observation": [
    {
      "similarity_score": 0.857,
      "content": {
        "label": "HC",
        "report": {
          "case_analysis": [
            "For ASD diagnosis using ABIDE fMRI, this network lacks key ASD biomarkers: no fronto-posterior disconnection
                 and no excessive local clustering in sensory regions.",
            "Social brain regions show normal integration, connecting to both DMN and executive networks.",
            "Default Mode Network proxies and Salience Network regions show balanced within- and between-network
                 connectivity, supporting typical anti-correlation dynamics.",
            "The network exhibits flexible modular architecture with well-defined but interconnected communities, unlike
                 ASD's reported overly rigid or fragmented modularity."
          ],
          "conclusion": [
            "The brain network displays neurotypical organization: efficient small-world architecture, balanced
                 integration via hub regions, and well-defined but interconnected functional communities.",
            "Key HC indicators include preserved fronto-temporal integration, symmetric inter-hemispheric connectivity,
                 absence of ASD-typical dysconnectivity patterns, and optimal modularity balance.",
            "Network topology aligns with HC classification from the ABIDE dataset, showing characteristic healthy
                 functional connectivity patterns."
          ]
        }
      }
    },
    {
      "similarity_score": 0.845,
      "content": {
        "label": "HC",
        "report": {
          "case_analysis": [
            "Compared to ASD literature showing DMN disruption, this network shows strong within-DMN-like connectivity in
                 association regions.",
            "The network shows absence of sensory hyper-connectivity patterns: visual and cerebellar regions maintain
                 appropriate segregation without excessive local or long-range connections.",
            "Preserved interhemispheric symmetry in homotopic connections contrasts with reported ASD reductions in
                 corpus callosum connectivity.",
            "Robust hub connectivity in association cortices contradicts ASD findings of weakened integrative hubs and
                 disconnection syndrome patterns."
          ],
          "conclusion": [
            "The brain network displays organizational features consistent with healthy neurotypical connectivity:
                 integrated higher-order networks, segregated sensory processing, and efficient small-world architecture.
                 ",
            "Network properties align with HC profiles from ABIDE studies, particularly balanced integration-segregation
                 and preserved interhemispheric coordination.",
            "Absence of connectivity anomalies commonly associated with ASD supports classification as HC."
          ]
        }
      }
    },
```

```
    {
      "similarity_score": 0.831,
      "content": {
        "label": "HC",
        "report": {
          "case_analysis": [
            "Compared to ASD networks in ABIDE studies, this graph shows balanced hub distribution, intact long-range
                  connections between frontal and posterior regions, preserved cerebellar integration, and dense
                  cingulate connectivity without fragmentation.",
            "The network shows no evidence of reduced fronto-posterior connectivity, excessive local clustering in
                  sensory regions, or salience network disintegration.",
            "The network topology exhibits efficient global integration and specialized local processing, matching HC
                  profiles."
          ],
          "conclusion": [
            "The functional connectivity network displays characteristics typical of healthy controls: balanced hub
                  architecture, integrated modular organization, preserved long-range connections, and efficient small-
                  world topology.",
            "Specific HC indicators include a high-degree parietal hub, intact frontal-parietal and limbic-temporal
                  integration, cerebellar cortical connectivity, and no evidence of ASD-related dysconnectivity patterns.
                  ",
            "While anatomical interpretations require AAL mapping, the topological features strongly align with normative
                   brain network organization observed in HC populations."
          ]
        }
      }
    },
    {
      "similarity_score": 0.828,
      "content": {
        "label": "HC",
        "report": {
          "case_analysis": [
            "The strong posterior DMN cluster connectivity contrasts with ASD findings of DMN hypoconnectivity in ABIDE
                  studies.",
            "Preserved homotopic symmetry aligns with HC patterns rather than ASD-associated interhemispheric
                  dysconnectivity.",
            "Moderate rather than excessive local clustering in sensory regions contradicts ASD hyperconnectivity
                  patterns in primary sensory areas.",
            "Well-distributed hub architecture supports efficient global integration typical of neurotypical networks.",
            "Absence of fragmented community structure or isolated modules is inconsistent with ASD network
                  disintegration phenotypes."
          ],
          "conclusion": [
            "The network exhibits properties consistent with healthy control architecture: modular organization,
                  efficient hubs, and balanced integration-segregation.",
            "Key features contradict ASD biomarkers: robust DMN connectivity, preserved symmetry, and absence of local
                  over-connection patterns.",
            "Graph metrics suggest typical small-world architecture rather than altered topology frequently observed in
                  ASD.",
            "While specific anatomical interpretations require AAL mapping validation, the overall connectivity pattern
                  strongly supports the HC classification.",
            "The binary, thresholded nature of this connectome limits detailed functional interpretation but shows clear
                  differentiation from ASD network phenotypes."
          ]
        }
      }
    },
    {
      "similarity_score": 0.824,
      "content": {
        "label": "HC",
        "report": {
          "case_analysis": [
            "Network shows strong long-range frontal-parietal-temporal connections, inconsistent with ASD hypo-
                  connectivity patterns.",
            "High integration of putative social brain regions suggests preserved social-cognitive network function.",
            "Balanced modularity is observed: distinct functional clusters maintain integration through hub nodes, unlike
                   ASD patterns of reduced between-module connectivity.",
            "Robust interhemispheric connectivity suggested by bilateral hub connections supports typical neural
                  coordination."
          ],
          "conclusion": [
            "The connectivity pattern exhibits hallmark features of healthy brain organization: efficient hub-mediated
                  integration, balanced modularity, and robust long-range connections.",
            "Topological properties, including small-world architecture, rich-club organization, and absence of
                  fragmentation, align with neurotypical resting-state networks.",
            "The network lacks ASD-typical signatures of reduced fronto-posterior connectivity or disrupted default-mode
                  integration, supporting the HC classification.",
            "While anatomical specificity is limited by binary AAL node data, the overall graph topology strongly
                  suggests healthy control connectivity patterns."
          ]
        }
      }
    }
  ]
}
```

J.3 Visualization of the Third-round Reasoning

The third-round reasoning result is shown below. After retrieving task-specific cases from the ABIDE case memory, BrainAgent summarizes the conflicting evidence from similar HC and ASD cases. Based on the updated report, the agent performs the final analysis and outputs the predicted label, supporting reasons, and confidence score.

Report-stage Reasoning Result (Round 3)


```
{
   "think": "The retrieved cases strongly indicate HC classification, noting absence of key ASD biomarkers: preserved
         fronto-posterior integration, robust DMN connectivity, balanced modularity, and no sensory hyperconnectivity.
         My initial analysis flagged 'typical ASD-related features' from input, but cases contradict this, suggesting
         the input's note may be misleading. Need to reconcile: input shows strong small-worldness, high modularity,
         and hub connectivityconsistent with HC per cases. Likely the input's 'ASD-related' comment is a red herring;
         network topology aligns with HC.",
   "report": "## Brain Network Analysis Report\n\n1. **Analyze**: Initial assessment reveals a brain network with
         strong small-world architecture , high modularity (Q=0.4696), densely connected posterior hubs, near-complete
         occipital-parietal cluster (density 0.92), moderate DMN connectivity, and fronto-subcortical integration.
         Input data mentions 'typical ASD-related features' but this appears inconsistent with detailed topology.\n2.
         **Action 1**: Retrieved 5 similar cases all labeled HC (similarity scores 0.857-0.824). Key observations:
         absence of ASD biomarkers-preserved fronto-posterior integration, robust DMN connectivity, balanced modularity,
          no sensory hyperconnectivity, and intact hub architecture. Cases indicate network exhibits neurotypical
         organization."
}
```


Final Analysis Reasoning Result (Round 3)


```
{
   "prediction": "HC",
   "reason": [
      "The network exhibits strong small-world architecture, which are hallmarks of efficient neurotypical brain
            organization, not the disrupted integration often seen in ASD.",
      "The posterior hub network (nodes 32, 56-59, 61-63, 84, 85) forms a densely connected core with high-degree nodes
            , indicating preserved hub integrity and efficient long-range communication typical of healthy controls.",
      "The near-complete occipital-parietal cluster (density 0.92) represents normal, highly synchronized sensory
            processing, contrasting with the sensory hyperconnectivity or hypocommunication often reported in ASD
            networks.",
      "Moderate Default Mode Network (DMN) intra-density (0.395) and preserved fronto-parietal integration (inter-
            density 0.106) align with neurotypical connectivity patterns, whereas ASD typically shows reduced long-
            range DMN connectivity and altered network segregation.",
      "Retrieved similar cases from the report all labeled HC (similarity scores 0.857-0.824) provide strong empirical
            evidence that this network topology is characteristic of healthy control brains in the ABIDE dataset."
   ],
   "confidence": 4
}
```


J.4 Visualization of the Reflection-stage Reasoning

The reflection-stage reasoning result is shown below. At this stage, the final analysis result, together with the intermediate reports and the original graph evidence, is fed back to the agent for self-verification and correction. The BrainAgent checks whether the final prediction and its supporting reasons are consistent with the graph topology and the previously retrieved evidence. In this example, the refined prediction remains `HC`, which is more reliable than direct inference.

Reflection-stage Reasoning Result


```
{
   "thinking": "I need to systematically analyze the LLM's output against the raw graph data and neuroscience knowledge.
          First, I'll examine the graph structure: there are 90 nodes (consistent with AAL atlas) and 242 edges with
         uniform weight 1.000. The LLM's analysis contains several issues: 1) It claims 'density = 0.2421\%' but with
         90 nodes, maximum possible edges = 4005 (n*(n-1)/2), actual density = 242/4005 = 6.04\%, not 0.2421\%. 2) It
         calculates modularity Q=0.4696 and small-world $\sigma$=3.0395 without showing methodology - these metrics
         require community detection algorithms and comparison to random networks. 3) It identifies 'DMN intra-density
         = 0.3949 vs DMN-FPN inter-density = 0.1062' but doesn't specify which nodes belong to DMN/FPN networks. 4) The
          prediction relies heavily on these unverified metrics. I need to perform my own analysis: Count node degrees
         to verify hub identification. Check connectivity patterns against known ASD neurobiology: ASD typically shows
         altered connectivity in default mode network, fronto-temporal connections, and reduced long-range integration.
          From the data: Nodes 32, 56-69, 84-89 form a highly interconnected cluster (degrees 15-19), likely
         representing default mode network regions. The network shows strong local clustering but also cross-hemisphere
          connections (e.g., nodes 0-1 connect to many bilateral regions). However, without knowing which AAL regions
         correspond to which nodes, I cannot verify anatomical claims. The LLM's confidence of 0.88 seems inflated
         given methodological issues. I should be more conservative.",
   "judgement of prediction": "0.65",
```

```
    "judgement of reason": "0.55",
    "reasons": [
        "The LLM's quantitative metrics (density, modularity, small-worldness) appear miscalculated or unsubstantiated -
            density is off by factor of 25, and other metrics lack calculation methodology.",
        "The reasoning assumes specific network assignments (DMN, FPN) without node-to-region mapping, making anatomical
            claims unverifiable.",
        "While the network shows high-degree hubs and clustering, ASD connectivity alterations are heterogeneous - some
            studies report hyperconnectivity in specific networks, others hypoconnectivity.",
        "The LLM correctly identifies strong small-world properties and hub structure, which generally favor HC
            classification, but overstates the certainty given data limitations.",
        "Without knowing which nodes correspond to which brain regions, the analysis of 'DMN dominance' and 'frontal-
            posterior connections' is speculative."
    ],
    "prediction_raw": "HC",
    "prediction_refined": "HC"
}
```


K Discussion

K.1 Findings and Take-away Messages

We highlight some key findings here, including some take-away messages from experimental results and findings during our practice.

Findings 1: Inherent Over-confident Phenomenon exist within mainstream LLMs.

We find that when directly using an LLM for inference, initial prompt design exerts a strong guiding effect on the model. For instance, explicitly asking LLM to conduct disease diagnosis in the prompt makes the model tend to give positive judgments. This tendency becomes even more pronounced in unfamiliar domains such as brain network analysis based on brain graphs. In some cases, the reasoning results remain completely consistent across multiple inference runs, as is shown in the identical outputs of *pass@1* and *pass@3*.

Findings 2: Reflection is a useful module in BrainAgent.

As is shown in comparison results and ablation studies, compared to other tricks like CoT, reflection takes a more positive role. In ablation studies, removal of reflection module leads to a relative great drop in metrics. These results are consistent with those reported in [30], which indicates that judging is an easier task for LLM than reasoning. Thus, after generation, prompting LLM to check and refine can reduce hallucination or factual errors to some extent.

Findings 3: In brain network analysis, LLMs yield little substantial performance gains over SLMs.

In our case, LLM's performance has little gain on SLMs (Small Language Models). We attribute this to this phenomenon in these aspects: First, compared to traditional reasoning tasks on pure text-modal, our direct conversion of graphs into textual format increases the difficulty for LLMs to understand the structural information of brain graphs to a certain extent. Second, task is highly domain-specific (brain network analysis), and the corresponding datasets are rarely included in the pre-training or post-training process of LLMs. This indicates that dedicated methods need to be designed for such specialized domains to enhance model's perception of domain knowledge. On the other hand, it also demonstrates that SLMs are capable of achieving performance comparable to LLMs on such relatively vertical and domain-specific tasks [1].

Findings 4: RAG focusing on specific data samples outperforms RAG for general knowledge.

Our ablation studies also suggest that CARD is more useful than HARK although both modules are proven effective for brain network analysis. We attribute this to that the reason lies in that although both modules aim to narrow the gap between general LLM and task-specific domains, the knowledge retrieved by CARD consists of sample-level data which are directly relevant to the target task. For HARK, however, it simply retrieves general knowledge, which may be less directly relevant to the target sample.

K.2 Unsuccessful Records

Although we design multiple modules and methods to ensure stable and accurate inference for BrainAgent, we admit that as LLM is a probability-based generative model, it is impossible that every trajectory is successful and exception during inference is inevitable. We list some cases indicating several typical errors for LLM generation which we think is meaningful for exploration of future LLM-based brain network analysis. Moreover, it does not imply that our BrainAgent is incapable of dealing with these issues.

Prediction mismatch may occur during LLM inference. Although we require LLM to output prediction from given two labels, we found that LLM may still output unexpected predictions.

An example from Claude prediction on ABIDE dataset.

UNCERTAIN - Likely ASD with moderate confidence (confidence: 3-4/10).

REASONING: While the LLM's analysis identifies a well-organized network with clear hub structure and modularity, several factors suggest this may be ASD rather than HC: .....

The LLM's HC prediction is PLAUSIBLE but OVERCONFIDENT.

As is shown in the above prediction from Claude, although we strictly require LLM to output only between "HC" and "ASD", Claude still outputs "UNCERTAIN" and list its reasons. On one hand, this illustrates that the model's instruction following ability cannot be fully maintained. On the other hand, it also implies that certain limitations still exist in current brain network analysis tasks, calling for design of more targeted and advanced benchmarks in neuroscience, which can be viewed as future work.

The ability to invoke customized tools exhibits inherent instability. In BrainAgent, information for tool invocation is directly delivered to the LLM in the form of prompts. We found several issues during BrainAgent's function calling:

1. LLM fails to invoke different tools across multiple rounds and only calls those placed at the forefront (in our case, LLM only calls `retrieve_knowledge` function). This can be regarded as "Attention Sink" phenomenon [12], where LLM tends to focus more on tokens at the forefront. In BrainAgent, we solve this issue by popping certain function from the function list after it is called. Thus, each function will be called once and only once.

2. Parameters of the function calls returned by LLM are inconsistent with the requirements specified in the prompt. For example, for function `retrieve_knowledge`, we define its parameter `query: str` and `regions_q: List[int]` indicating query content and brain regions respectively. However, for LLM generated json which is shown in K.2, its arguments become `regions_t` and `query_t` which can cause exception when parsing LLM outputs.

An example from Qwen 3.5. (HARK module)

{
"action": "retrieve_knowledge",
"args": {
"regions_t": ["2", "4", "5", "8", "9", "11", "19",],
"query_t": "Weighted undirected fMRI functional connectivity graph....."
}
}

In conclusion, most unsuccessful trials can be viewed as instruction following issues, and we think it can be solved by supervised fine-tuning (SFT) or Reinforcement Learning (RL), which will be discussed in Section L.

K.3 Ethical Statements

All the datasets used in BrainAgent are public datasets, thus there is no ethical concern for privacy information leakage. We strictly adhered to their respective usage agreements. All preprocessing pipelines follow official procedures provided by the dataset maintainers.

We discuss more about possible negative social impacts. As part of the research in this paper deals with the diagnosis of depression, it is necessary to elaborate here on the possible negative social impacts of this work, despite the fact that all the current work is at the stage of scientific research and has not been put to practical use. Including but not limited to:

- **Incorrect diagnosis**. AI methods must have the possibility of error, which cannot be avoided, but an incorrect diagnosis will have a significant impact on individuals and society. Therefore, AI tools can only be used as a diagnostic aid, not as a decision maker, and the final decision should still be made by the doctor.
- **Role of BrainAgent in real-world diagnosis.** Finally we position our method as an assistant rather than direct decision maker for doctors. We fully acknowledge that many of these complexities must be addressed before real-world deployment; however, covering every contingency is, at present, beyond the reach of any single AI approach. We therefore contend that a more effective and safer strategy is to treat our AI diagnostic model as an auxiliary instrument. These outputs from BrainAgent inform rather than override doctors' final decision. In short, while our work makes a constructive exploration in improving downstream performance and interpretability for directly using LLM in neuroscience task, it is ultimately designed to relieve human's labor and assist rather than replace human medical judgment.

L Limitations and Future Works

BrainAgent is a novel agentic framework which directly use LLM as predictor in enhanced and interpretable brain network analysis, and comprehensive experiments on diverse public real-world datasets proved its effectiveness. Despite these promising results, BrainAgent still faces several limitations, which are outlined below:

- **Instruction following capabilities and tool use**: BrainAgent still faces instruction following issues on its structure output, with its tool invocation procedure also being relatively fixed. Here, we solve these issues through relatively straightforward engineering tricks. In future work, such limitations can be intrinsically addressed by finetuning the model and designing specific post-training paradigms to regulate the inherent preferences of LLM.
- **Hyper-parameter selection**: BrainAgent is a novel framework composed of multiple modules, with each module involving several certain hyperparameters. Most of these parameters are set to fixed values in our implementation, yet we found the results are already promising. We believe that performance gains of BrainAgent could be further improved through hyperparameter searching. Nevertheless, our primary focus lies in the design of the overall pipeline. Furthermore, We attribute this to that compared to pure hyperparameter searching, dataset curation and procedural optimization are considered more decisive factors.
- **Scaling up and improving knowledge base**: The construction of datasets and knowledge bases is a labor-intensive work which requires substantial human labor. In our paper, knowledge base in HARK and brain graph dataset in CARD are maintained at a relatively modest scale (2000 cases for HARK), which constrains the further enhancement of BrainAgent's performance to some extent. For future research,expanding the scope of literature data sources can be considered, as well as designing more rigorous and comprehensive pipelines for data curation and validation, and adopting a hybrid verification paradigm integrating LLM-Human verification, thereby improving both the scale and quality of the knowledge base and dataset.

We also list some challenges which can also be viewed as future work:

- **Designing benchmarks tailored for LLM-based brain network analysis**: BrainAgent is a first attempt to utilize LLM as predictor, yet its main task is brain network classification

and analysis. Designing specific and challenging benchmarks to comprehensively evaluate LLM's ability in neuroscience field is a meaningful topic for building LLM suited for brain network scenarios.

- **Improving token efficiency**: BrainAgent is an agentic framework which requires multi-round iteration. Although it is useful, especially in improving accuracy and interpretability, as well as reducing hallucination compared to direct inference, yet we admit that it leads to more token consumption. Here we utilize several methods to alleviate this issue like maintaining a fixed-size report and using understanding outputs for iteration instead of raw data, we think designing more efficient or parallel ways to improve LLM's efficiency is also a promising direction.
- **Designing Advanced agentic pipelines**: As there is no previous work, we design two RAG modules (HARK and CARD), aiming to inject domain and task specific knowledge into LLMs. For future work, it is advisable to design more complex and task-specific agent workflows, along with more RAG and knowledge injection approaches. Meanwhile, our BrainAgent is a training-free network, and its LLM core can be substituted by open-source models which can be finetuned on neuroscience datasets, thereby improving model's internal awareness of specific field.
- **Internalizing LLM's ablility into SLM for practical deployment and usage**: Here we use public datasets from real world to illustrate BrainAgent's effectiveness. In clinical scenes, cases are more complex which requires privacy protection for patients' personal information, which means all the inference will be run locally. On the other hand, our comparison results also suggest that for specific task, SLMs have the potential to achieve performance comparable to, or even superior to that of LLMs. Therefore, designing post-training paradigms, as well as other compression methods like distillation to internalize these complex analytical skills into SLMs which requires less GPU memory, can significantly reduce the resource consumption for local deployment requirements while maintaining their performance comparable to that of LLMs, thus promoting the feasibility and efficiency of practical deployment, implementation and usage.