

A MORE DISCUSSION ABOUT MULTI-ATLAS

A.1 DIFFERENCE BETWEEN MULTI-ATLAS AND MULTI-TEMPLATE METHODS

In brain MRI analysis, the concepts of template and atlas can sometimes be confused. The atlas in our paper refers to a detailed map of brain structures (Tzourio-Mazoyer et al., 2002), often derived from anatomical, functional, and histological data, and includes labels for different brain regions based on specific criteria. A template, on the other hand, is a standard reference image that serves as a common coordinate system for comparing different brain images (Tang et al., 2016). It is usually created by averaging brain images from a group of subjects, providing a standardized space for registering or aligning individual brain images. This allows for comparison and combination of data across different subjects or groups. Figures 3 and 4 illustrate the difference between atlas and template. Although some existing works (Min et al., 2014a;b; Liu et al., 2015; 2016) are named multi-atlas, they are more akin to multi-template methods. Instead of registering brain images to different spaces in multi-template methods, the multi-atlas methods discussed in our paper segment brain images in the common space to define ROIs differently.

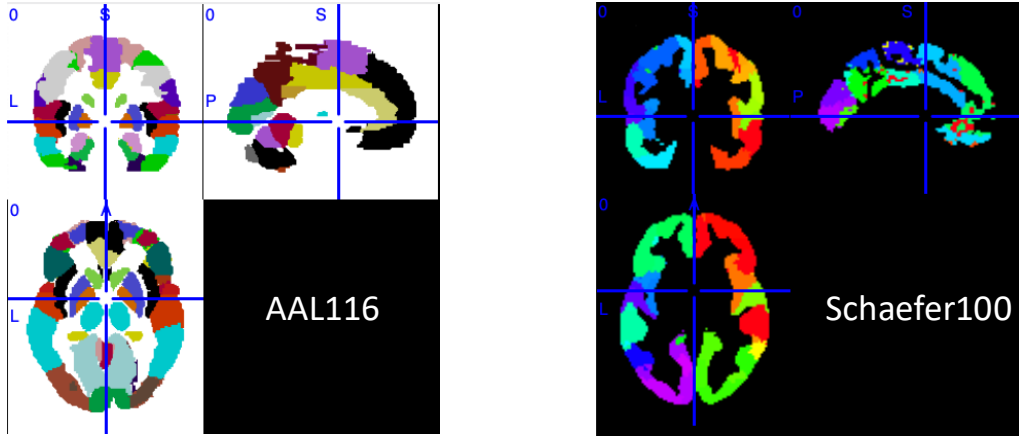


Figure 3: AAL116 and Schaefer100 atlases. Each atlas is based on a different parcellation hypothesis.

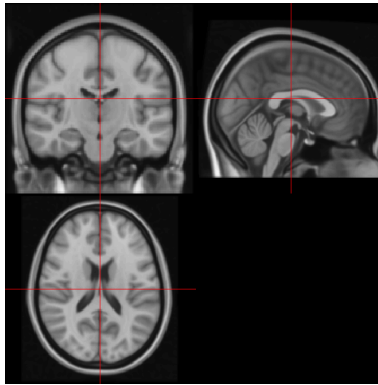


Figure 4: MNI template T1-w image.

A.2 DIFFERENCES OF VARIOUS ATLASES

When applying an atlas to preprocessed neuroimaging data, not all voxels are assigned to a specific ROI based on the distinct hypothesis for different atlases. For instance, in our study, 6.3% of voxels

in the Schaefer atlas are not included in AAL; 33.3% of voxels in AAL are not included in Schaefer; 13.8% of voxels in the HO atlas are not included in Schaefer; and 13.8% of voxels in Schaefer are not included in HO. This incomplete ROI assignment can result in the loss of important information for downstream tasks. A figure comparing AAL and Schaefer atlases is provided in the Rebuttal PDF, showing that each atlas is based on a different parcellation hypothesis with significantly different ROI definitions. For example, Schaefer does not include the whole cerebellum, unlike AAL. We summarize the detailed comparison of AAL and Schaefer atlases in the following:

- **Development**

AAL Atlas: Introduced by Tzourio-Mazoyer et al. (2002).

Schaefer Atlas: Created by Schaefer et al. (2018b).

- **Basis**

AAL Atlas: Based on anatomically defined regions.

Schaefer Atlas: Based on functional connectivity data.

- **Number of Regions**

AAL Atlas: 116 regions.

Schaefer Atlas: Variable resolutions, offering 100, 200, 300, up to 1000 parcels.

- **Coverage**

AAL Atlas: Covers the entire cerebral cortex, subcortical structures, and cerebellum.

Schaefer Atlas: Covers the entire cerebral cortex only.

- **Resolution**

AAL Atlas: Coarse, anatomically-based.

Schaefer Atlas: Multi-resolution, functionally-based.

- **Primary Use**

AAL Atlas: Commonly used in functional and structural neuroimaging studies.

Schaefer Atlas: Suitable for high-resolution functional parcellation and network-based analyses.

- **Defining Features**

AAL Atlas: Features straightforward anatomical definitions and ease of use.

Schaefer Atlas: Captures functional organization with higher precision.

- **Strengths**

AAL Atlas: Simple, established, and widely used in the neuroimaging community.

Schaefer Atlas: Provides detailed functional parcellations and flexibility in resolution.

- **Limitations**

AAL Atlas: Limited to anatomical boundaries and offers coarse granularity.

Schaefer Atlas: Requires functional data and is more complex to implement.

- **Typical Applications**

AAL Atlas: General neuroimaging studies.

Schaefer Atlas: Advanced fMRI research and network analysis.

B NOTATION TABLE

Notation-wise, we use calligraphic letters to denote sets (e.g., \mathcal{X}), bold capital letters to denote matrices (e.g., \mathbf{X}), and strings with bold lowercase letters to represent vectors (e.g., \mathbf{x}). Subscripts and superscripts are used to distinguish between different variables or parameters, and lowercase letters denote scalars. We use $\mathbf{S}[i, :]$ and $\mathbf{S}[:, j]$ to denote the i -th row and j -th column of a matrix \mathbf{S} , respectively. Table 6 summarizes the notations used throughout the paper.

C MORE DISCUSSION ABOUT THE IDENTITY EMBEDDING AND POSITIONAL EMBEDDING

The identity and positional information in our context of ROIs differ from each other. In graph Transformers, positional embeddings (e.g., graph Laplacian) (Li et al., 2020; Ying et al., 2021;

Table 6: Notation table

Notation	Description
\mathcal{D}	Input dataset
$\mathbf{X}_a, \mathbf{X}_b$	A brain network of atlases a and b
y_X	Label of \mathbf{X}_a and \mathbf{X}_b
n_a, n_b	Number of nodes in atlases a and b
\mathbf{H}_{ID}	Identity-encoded feature matrix
\mathbf{m}	Brain network representation
d	Dimensionality of node representations
$\mathbf{W}_{ID}, \mathbf{W}_{INC}, \mathbf{W}_{GC}, \mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V$	Parameter matrices
\mathbf{H}	Node representations after inter-atlas message-passing
\mathbf{A}^{ab}	The cross-atlas adjacency matrix for atlases a and b
\mathbf{H}^{ab}	The combined node representations for atlases a and b
i, j	Index for matrix dimensions
bz	Batch size

Wang et al., 2022) are designed to encode graph structural information. However, in brain networks where edges are defined by the correlation between ROIs, the network can be considered a fully connected, weighted graph. Thus, no structural information can be captured by general positional embeddings. Furthermore, the orders of ROIs in different brain networks with the same atlas are identical. **Therefore, we use identity embeddings to mark nodes in different subjects that correspond to the same ROI. To be more specific, the first node in all brain networks with AAL atlas represents the same ROI. We add the identity embedding of this ROI ($\mathbf{W}_{ID}[1, :]$) to its node feature $\mathbf{X}[i, :]$.** The vectorized embedding kernel in BrainGNN (Li et al., 2021) and our identity embedding both aim to encode identity information into node representations, but ours is a simpler way without relying on the spatial information.

D MORE DETAIL ABOUT INTER-ATLAS MESSAGE-PASSING

D.1 DETAILED DESCRIPTION OF INTER-ATLAS MESSAGE-PASSING.

Given two atlases a and b with n^a and n^b ROIs, we denote the 3D coordinate of the i -th and j -th ROI of them as \mathbf{C}_i^a and \mathbf{C}_j^b , respectively. The distance matrix $\mathbf{D}^{ab} \in \mathbb{R}^{n_a \times n_b}$ is computed by Euclidean distance $\mathbf{D}_{ij}^{ab} = \text{distance}(\mathbf{C}_i^a, \mathbf{C}_j^b)$.

A mask matrix \mathbf{M} is then generated, where $\mathbf{M}_{ij}^{ab} = 1$ if $j \in \text{topk}(\mathbf{D}_i^{ab})$, and 0 otherwise. Afterwards, the inter-atlas adjacency matrix is defined as: $\mathbf{A}^{ab} = \begin{bmatrix} 0 & \mathbf{M}^{ab} \\ \mathbf{M}^{ba} & 0 \end{bmatrix}$.

We summarize this process in the following algorithm.

Algorithm 1 Construction of the inter-atlas adjacency matrix \mathbf{A}^{ab} .

Require: The 3D ROI coordinates \mathbf{C}^a and \mathbf{C}^b of atlases a and b ;

- 1: $\mathbf{D}_{ij}^{ab} = \text{distance}(\mathbf{C}_i^a, \mathbf{C}_j^b)$;
 - 2: $\mathbf{M}_{ij}^{ab} = \begin{cases} 1 & \text{if } j \in \text{topk}(\mathbf{D}_i^{ab}); \\ 0 & \text{otherwise} \end{cases}$;
 - 3: $\mathbf{D}_{ij}^{ba} = \text{distance}(\mathbf{C}_i^b, \mathbf{C}_j^a)$;
 - 4: $\mathbf{M}_{ij}^{ba} = \begin{cases} 1 & \text{if } j \in \text{topk}(\mathbf{D}_i^{ba}); \\ 0 & \text{otherwise} \end{cases}$;
 - 5: $\mathbf{A}^{ab} = \begin{bmatrix} 0 & \mathbf{M}^{ab} \\ \mathbf{M}^{ba} & 0 \end{bmatrix}$;
 - 6: **return** \mathbf{A}^{ab}
-

D.2 AN EXAMPLE OF THE ADJACENCY MATRIX FOR INTER-ATLAS MESSAGE-PASSING

As shown in Figure 5, we generated an adjacency matrix for Schaefer100 and AAL116 by setting $k = 5$. The connections between spatial neighborhoods across atlases are constructed. Note that node v in atlas 1 is one of the nearest k neighbors of node u in atlas 2 does not mean u is one of the nearest k neighbors of v , thus the adjacency matrix is asymmetry.

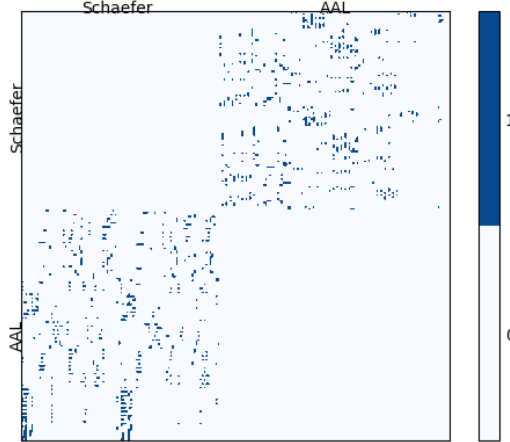


Figure 5: The adjacency matrix for inter-atlas message-passing.

E DETAILED DATASET DESCRIPTION

The class-wise sample sizes are summarized in Table 7.

ABIDE The ABIDE initiative supports the research on ASD by aggregating functional brain imaging data from laboratories worldwide. ASD is characterized by stereotyped behaviors, including irritability, hyperactivity, depression, and anxiety. Subjects in the dataset are classified into two groups: TC and individuals diagnosed with ASD.

ADNI The ADNI raw images used in this paper were obtained from the ADNI database (adni.loni.usc.edu). The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging (MRI), PET, other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early AD. For up-to-date information, see www.adni-info.org. We include subjects from 6 different stages of AD, from cognitive normal (CN), significant memory concern (SMC), mild cognitive impairment (MCI), early MCI (EMCI), late MCI (LMCI) to AD.

PPMI The PPMI is a comprehensive study aiming to identify biological markers associated with Parkinson’s risk, onset, and progression. PPMI comprises multimodal and multi-site MRI images. The dataset consists of subjects from 4 distinct classes: normal control (NC), scans without evidence of dopaminergic deficit (SWEDD), prodromal, and PD.

Mātai Mātai is a longitudinal single site, single scanner study designed for detecting subtle changes in the brain due to a season of playing contact sports. This dataset consists of the brain networks preprocessed from the data collected from Gisborne-Tairāwhiti area, New Zealand, with 35 contact sport players imaged at pre-season ($N=35$) and post-season ($N=25$) with subtle brain changes confirmed using diffusion imaging study due to playing contact sports.

F IMPLEMENTATION DETAILS

The settings of our experiments mainly follow those in (Dwivedi et al., 2020). We split each dataset into 8:1:1 for training, validation and test, respectively. We evaluate each model with the same

Table 7: The Class Distribution of the Brain Network Datasets we used

Dataset	Gender (F/M)	Age (mean \pm std)	Class	# Subjects
ABIDE	152/873	16.5 \pm 7.4	Control	537
			ASD	488
			CN	819
			SMC	73
ADNI	728/599	74.6 \pm 7.9	LMCI	102
			EMCI	89
			MCI	179
			AD	65
			Control	15
PPMI	82/127	62.9 \pm 9.5	SWEDD	14
			Prodromal	67
			PD	113
Mātai	N/A	N/A	Pre-season	35
			Post-season	25

random seed under 10-fold cross-validation and report the average accuracy. The whole network is trained in an end-to-end manner using the Adam optimizer (Kingma & Ba, 2014). We use the early stopping criterion, i.e., we halve the learning rate when there is no further improvement on the validation loss during 25 epochs and stop the training once the learning rate is smaller than the minimum rate we set. All the codes were implemented using PyTorch (Paszke et al., 2017) and Deep Graph Library (Wang et al., 2019) packages. All experiments were conducted on a Linux server with an AMD Ryzen Threadripper PRO 5995WX 64-Cores and an NVIDIA GeForce RTX 4090. The version for the software we used in AIDFusion is listed in Table 8.

Table 8: The software dependency of AIDFusion.

Dependency	Version
Python	3.10.13
cuda toolkit	12.2
pytorch	2.2.1+cu121
DGL	2.1.0+cu121
scikit-learn	1.4.1.post1
numpy	1.26.4
matplotlib	3.8.3
nilearn	0.10.4

All the baselines used in this paper are implemented by ourselves. In AIDFusion, we adopt a mean pooling layer as the readout function and a two-layer MLP with ReLU as the prediction head. The number of incompatible nodes r is set to 4. The number of clusters n' for the cluster assignment matrix S is set to half of the average number of nodes for all atlases, e.g., for Schaefer100 and AAL116, $n' = (100 + 116)/2/2 = 54$. The temperature hyper-parameter τ in Eq. (10) is set to 0.75. The hidden dimensions of all layers are set to 100. The batch size of each dataset is set to 10% of the subject number in the dataset. We tuned the other hyperparameters on validation set including k in k NN for the inter-atlas message-passing, trade-off hyperparameters $\lambda_1, \lambda_2, \lambda_4, \lambda_4$ for loss function Eq. (12), the initial learning rate $init_lr$, and the minimum learning rate min_lr used for early stop. To be specific, we search k from $\{3, 5, 10\}$, λ_1 from $\{1e-2, 1e-1, 1e0, 1e1, 1e2\}$, λ_2 from $\{1e-2, 1e-1, 1e0, 1e1, 1e2\}$, λ_3 from $\{1e-6, 1e-5, 1e-4, 1e-3, 1e-2\}$, λ_4 from $\{1e-1, 1e0, 1e1\}$, $init_lr$ from $\{5e-5, 8e-5, 1e-4, 2e-4, 7e-4\}$ and min_lr from $\{6e-5, 1e-5, 5e-6, 1e-6\}$. The optimized hyperparameters for AIDFusion are reported in Table 9.

For machine learning baselines (LR, SVM), we use the official code¹ of Xu et al. (2023) and implement the multi-atlas version on the top of it. We follow the same parameter-tuning scheme as they do. The full list of parameters, including those we used for the grid search, is provided below. The parameter tuning for MultiLR and MultiSVM are the same with LR and SVM, respectively.

¹https://github.com/brainnetuoa/data_driven_network_neuroscience

Table 9: The optimized hyperparameters for AIDFusion.

	ABIDE	ADNI	PPMI	Mātai
k	10	5	5	5
λ_1	1e1	1e1	1e1	1e-2
λ_2	1e-1	1e1	1e1	1e0
λ_3	1e-2	1e-5	1e-4	1e-3
λ_4	1e0	1e0	1e0	1e0
$init_lr$	1e-4	8e-5	1e-4	7e-4
min_lr	6e-5	6e-5	6e-5	6e-5

LogisticRegression(*penalty*:('l1', 'l2', 'elasticnet', 'none'), *dual*=False, *tol*=0.0001, *C*=1.0, *fit_intercept*=True, *intercept_scaling*=1, *class_weight*=None, *random_state*=None, *solver*='lbfgs', *max_iter*=1000000, *multi_class*='auto', *verbose*=0, *warm_start*=False, *n_jobs*=None, *l1_ratio*=None)

SVM(*C*=1, *kernel*=('rbf', 'linear', 'poly', 'sigmoid', 'shortest path'), *degree*=3, *gamma*='auto', *coef0*=0.0, *shrinking*=True, *probability*=False, *tol*=0.001, *cache_size*=200, *class_weight*=None, *verbose*=False, *max_iter*=-1, *decision_function_shape*='ovr', *break_ties*=False, *random_state*=None)

Table 10: Results of more atlases on ADNI dataset. The best results for each atlas setting are highlighted in **bold**.

Atlas			model	acc \pm std
Schaefer100	AAL116	HO48		
✓	✓		MGRL	62.74 \pm 3.55
			MGT	63.99 \pm 4.34
			METAFormer	66.52 \pm 2.63
			LeeNet	64.63 \pm 1.34
			AIDFusion	67.57 \pm 2.04
	✓	✓	MGRL	64.48 \pm 1.68
			MGT	60.48 \pm 1.91
			METAFormer	64.48 \pm 1.68
			LeeNet	64.48 \pm 1.58
			AIDFusion	65.99 \pm 2.62
✓		✓	MGRL	57.92 \pm 2.82
			MGT	60.64 \pm 3.52
			METAFormer	65.23 \pm 3.25
			LeeNet	63.42 \pm 1.82
			AIDFusion	65.91 \pm 1.80
✓	✓	✓	MGRL	56.63 \pm 4.66
			MGT	62.89 \pm 1.28
			METAFormer	66.33 \pm 2.80
			LeeNet	64.40 \pm 1.71
			AIDFusion	66.59 \pm 1.77

G RESULTS WITH MORE ATLASES

To further evaluate the effectiveness of AIDFusion with other atlases, we conducted experiments using an additional atlas, HO48 (Makris et al., 2006), on the ADNI dataset. **Note that when AIDFusion is extended to handle more than two atlases, it computes the orthogonal loss, inter-atlas message passing, and both subject-level and population-level consistency for each pair of atlases.** The results, presented in Table 10 indicate that the proposed AIDFusion achieves the best performance across all four atlas settings. Notably, increasing the number of atlases does not necessarily enhance model performance. In some cases, using all three atlases yields lower accuracy compared to the combination of Schaefer100 and AAL116. Additionally, the choice of atlas combination is crucial for

multi-atlas methods. In dual-atlas experiments, combining two atlases with a similar number of ROIs (Schaefer100 and AAL116) can mutually enhance and significantly improve model performance. To further verify our finding that our method will benefit more from atlases with a similar number of ROIs, we conduct additional experiments on the ABIDE dataset using three atlases (Schaefer100, AAL116, and BASC122 (Bellec et al., 2010)). Results shown in the following table demonstrate that for five multi-atlas methods, three of them (MGRL, METAFormer, and AIDFusion) achieve better performance with three atlases compared to any two-atlas combinations. Importantly, our proposed AIDFusion still outperforms all baselines in these settings. This reinforces our claim that AIDFusion effectively integrates multi-atlas information while remaining robust to the inclusion of additional atlases.

Table 11: Results of more atlases on ABIDE dataset. The best results for each atlas setting are highlighted in **bold**.

	Atlas			
Schaefer100	AAL116	BASC122	model	acc \pm std
			MGRL	61.56 \pm 4.90
			MGT	63.32 \pm 3.90
✓	✓		METAFormer	61.27 \pm 4.05
			LeeNet	61.28 \pm 3.12
			AIDFusion	66.35 \pm 3.26
			MGRL	60.80 \pm 5.12
			MGT	58.49 \pm 5.64
	✓	✓	METAFormer	62.92 \pm 5.79
			LeeNet	60.01 \pm 4.00
			AIDFusion	65.97 \pm 3.60
			MGRL	63.14 \pm 4.17
			MGT	59.90 \pm 3.46
✓		✓	METAFormer	62.53 \pm 3.78
			LeeNet	59.12 \pm 4.20
			AIDFusion	64.79 \pm 2.80
			MGRL	63.62 \pm 4.57
			MGT	62.84 \pm 3.85
✓	✓	✓	METAFormer	65.80 \pm 5.61
			LeeNet	60.19 \pm 3.77
			AIDFusion	66.65 \pm 4.14

To further explore how the resolution of atlases will influence the performance of our model, we conduct experiments for atlases with various numbers of ROIs. The Schaefer atlas allows adjusting the resolution of ROIs (e.g., from 100 to 1000). We selected Schaefer100 for detailed study because a previous study (Xu et al., 2023) found that using 100 ROIs with the Schaefer atlas usually performs better than using more ROIs. To verify this conclusion in multi-atlas brain network classification, we conducted experiments using AAL116 combined with Schaefer200, Schaefer500, and Schaefer1000. Results showed that AAL116 combined with Schaefer100 achieves the best results. Additionally, we provided experimental results using the HO48 atlas in Appendix F. The dual-atlas experiments demonstrate that combining two atlases with a similar number of ROIs (AAL116 and Schaefer100) can mutually enhance and significantly improve model performance. It is also interesting for us to explore using both atlases of around 200 nodes (or around 1000 nodes). We will leave such exploration about multi-scale brain networks in the future.

Table 12: Results of more atlases with different resolutions on ABIDE dataset. The best results for each atlas setting are highlighted in **bold**.

Atlas 1	Atlas 2	acc \pm std
AAL116	Schaefer100	66.35 \pm 3.26
AAL116	Schaefer200	65.03 \pm 5.10
AAL116	Schaefer500	64.72 \pm 4.80
AAL116	Schaefer1000	63.15 \pm 2.80

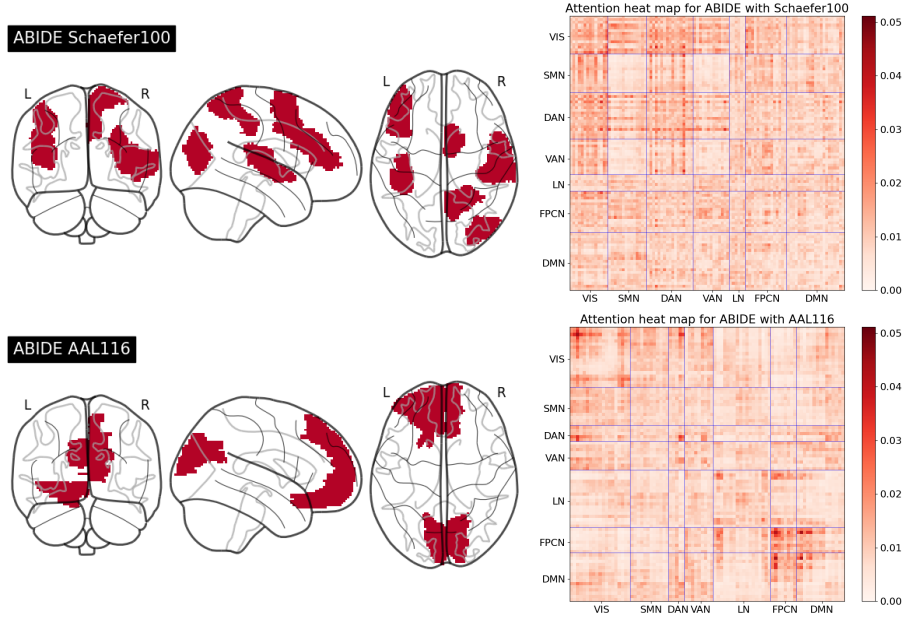


Figure 6: Visualization for attention maps on ABIDE. VIS = visual network; SMN = somatomotor network; DAN = dorsal attention network; VAN = ventral attention network; LN = limbic network; FPCN = frontoparietal control network; DMN = default mode network.

H BRAIN NETWORK ATTENTION MAP ON ABIDE

In the ASD analysis using the ABIDE dataset (Figure 6), AIDFusion identifies common connections within the lingual gyrus of the VIS network in both Schaefer100 and AAL116. This aligns with existing ASD studies which suggest a greater reliance on visual perceptual processing and more effortful top-down control during semantic processing in ASD (Shen et al., 2012). Besides, for Schaefer, AIDFusion emphasizes ROIs in the DAN, particularly the connection between the right posterior and VIS, consistent with Koshino et al. (2008) who found lower activation in ASD subjects in the inferior left prefrontal area (verbal processing and working memory) and the right posterior temporal area (theory of mind processing). In the AAL analysis, connections in the DMN and the frontoparietal control network (FPCN) are highlighted, supporting the understanding that (1) dysfunctions in DMN nodes and their interactions contribute to difficulties of ASD in attending to socially relevant stimuli (Padmanabhan et al., 2017), and (2) the ASD group shows reduced lateral frontal activity and diminished hippocampal connectivity, especially between the hippocampus and FPCN regions (Cooper et al., 2017). These findings clarify why the features identified by AIDFusion are distinctive for ASD biomarkers.

I MORE RESULTS FOR ABLATION STUDY

I.1 VISUALIZATION FOR THE ABLATION ON SUBJECT- AND POPULATION-LEVEL CONSISTENCY

In this subsection, we assess the impact of subject- and population-level consistency constraints on the representation learning of AIDFusion.

First, to understand the effect of the subject-level consistency loss, we compute the difference in hidden feature matrices \hat{H} for two different brain atlases: Schaefer100 and AAL116. We compare AIDFusion models with and without the subject-level consistency loss by plotting the difference matrix $\Delta\hat{H} = \hat{H}_{Schaefer100} - \hat{H}_{AAL116}$. As shown in Figure 7, incorporating the subject-level consistency loss significantly reduces the differences in hidden feature matrices across the two atlases. This demonstrates that AIDFusion achieves more consistent feature representations, aligning

the learned embeddings across different brain network views at a higher hidden layer. The reduced variance in the difference matrix implies that the model successfully learns shared patterns across atlases, leading to improved representation stability.

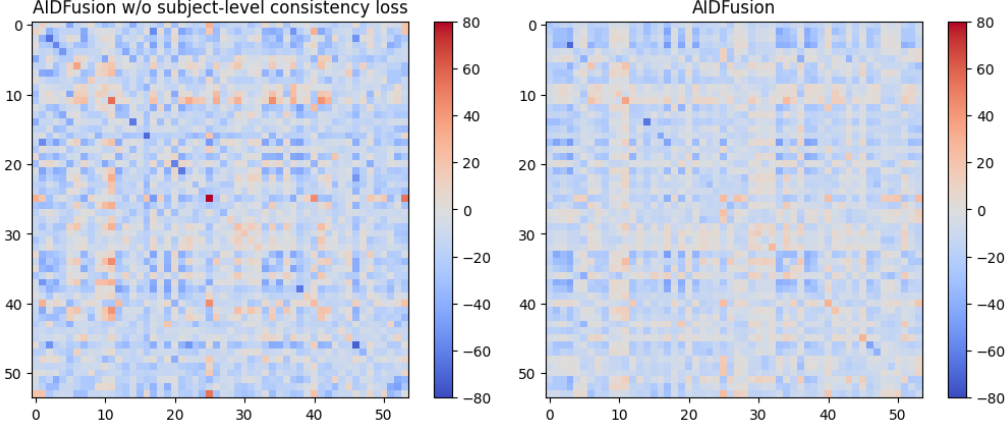


Figure 7: Visualization for the difference of hidden feature matrices \hat{H} between Schaefer100 and AAL116 on ADNI. The right one is AIDFusion while the left one is AIDFusion w/o subject-level consistency loss.

Next, we evaluate the effect of the population-level consistency loss by analyzing the difference in similarity matrices G (used in Eq. (11)), which capture the pairwise similarity of subject representations for Schaefer100 and AAL116 atlases. Figure 8 illustrates the difference in similarity matrices between the two atlases for models trained with and without the population-level consistency loss. Without this constraint (left panel), some subjects show significant discrepancies between the two views, suggesting that the learned representations are inconsistent across atlases. This misalignment may lead to degraded model performance, as the decision boundaries are influenced by inconsistent representations. In contrast, when applying the population-level consistency loss (right panel), the differences in similarity matrices are notably reduced. This indicates that AIDFusion successfully aligns the representations at the population level, ensuring that the graph-level similarities are preserved across different views of the brain network. By enforcing this consistency, our model can make more robust predictions.

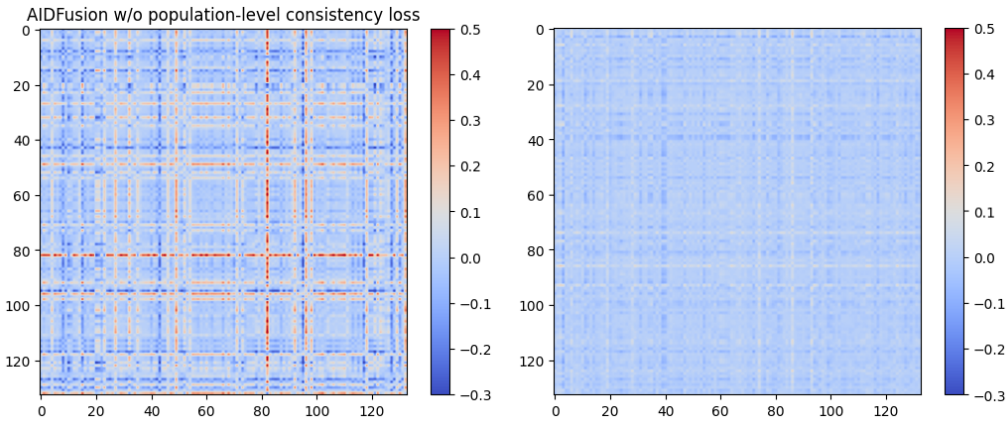


Figure 8: Visualization for the difference of the similarity matrices G between Schaefer100 and AAL116 on the first batch of ADNI. The right one is AIDFusion while the left one is AIDFusion w/o population-level consistency loss.

Overall, these visualizations highlight the importance of the subject- and population-level consistency losses in improving the alignment and robustness of the learned representations in multiview brain network analysis.

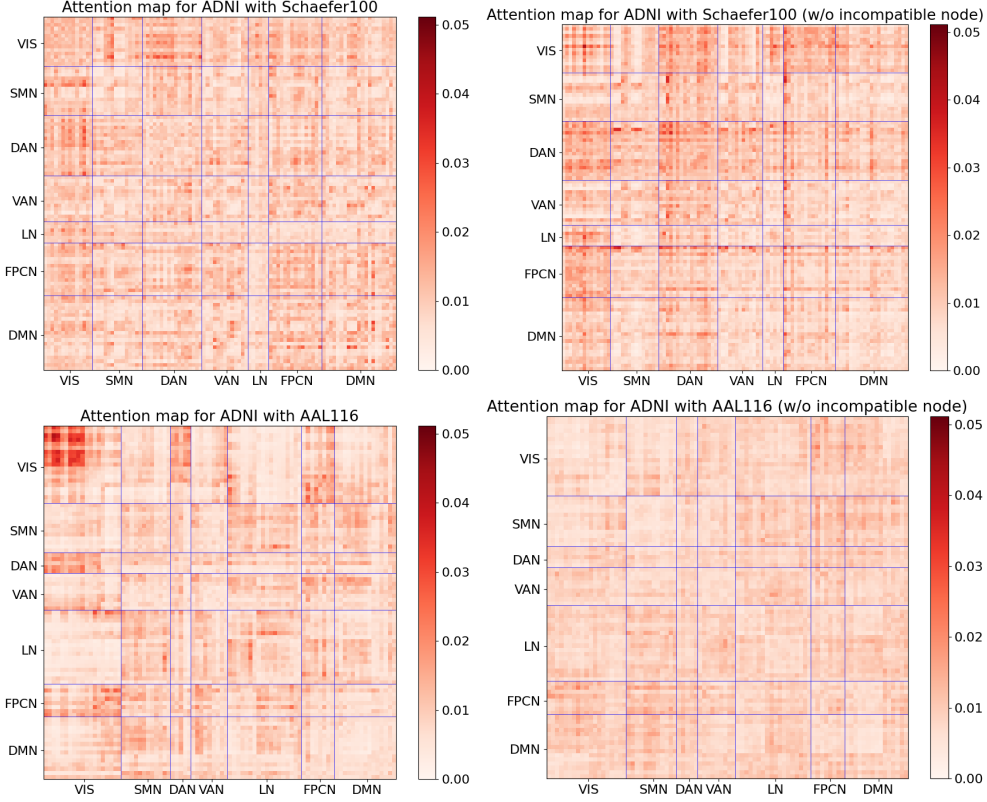


Figure 9: Visualization for attention maps of AIDFusion w/ and w/o incompatible nodes.

I.2 ABLATION STUDY FOR INCOMPATIBLE NODES

In this subsection, we further explore the function of incompatible nodes by visualizing the attention map of AIDFusion w/o incompatible nodes. The attention maps are shown in Figure 9. We can observe that, when not using incompatible nodes, the attentions of two atlases (in the right column) are remarkably imbalanced. Attentions on Schaefer are much higher than those on AAL. Besides, the attention map of AAL exhibits over-smoothing and no highlighted network is found, which indicates the model is not able to extract the distinguishable connections. This case study demonstrates that the incompatible nodes enable the model to filter out the inconsistent atlas-specific information.

J HYPERPARAMETER ANALYSIS

In this section, we study the sensitivity of four trade-off hyperparameters in Eq. (12). All experiments are conducted on the ADNI dataset. We tune the value of λ_1 from $1e0$ to $1e2$, λ_2 from $1e0$ to $1e2$, λ_3 from $1e-6$ to $1e-4$ and λ_4 from $1e-1$ to $1e1$. The results presented in Table 13 show that our model performs the best when $\lambda_1 = 1e1$, $\lambda_2 = 1e1$, $\lambda_3 = 1e-5$ and $\lambda_4 = 1e0$. We can exhibit that these trade-off hyperparameters in the loss function will marginally affect the model performance on ADNI (less than 1%), which demonstrates the stability of AIDFusion.

Table 13: The hyperparameter sensitivity analysis for AIDFusion on ADNI dataset.

λ_1	λ_2	λ_3	λ_4	acc \pm std
1e0	1e1	1e-5	1e0	66.97 \pm 1.95
1e1	1e0	1e-5	1e0	66.44 \pm 2.88
1e1	1e1	1e-6	1e0	67.04 \pm 2.20
1e1	1e1	1e-5	1e-1	66.82 \pm 1.98
1e1	1e1	1e-5	1e0	67.57 \pm 2.04
1e1	1e1	1e-5	1e1	66.82 \pm 2.64
1e1	1e1	1e-4	1e0	67.04 \pm 2.21
1e1	1e2	1e-5	1e0	66.89 \pm 2.10
1e2	1e1	1e-5	1e0	66.21 \pm 2.34