FAVA: FREQUENCY ALIGNMENT ACROSS VIEWS FOR MULTIVIEW GRAPH DOMAIN ADAPTATION

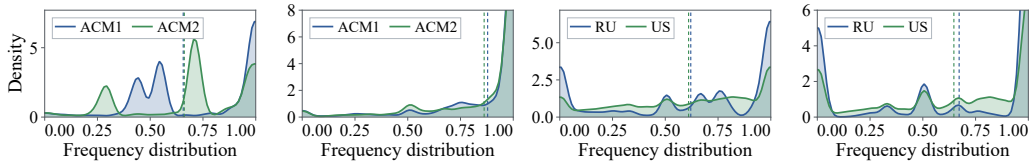
Anonymous authors
Paper under double-blind review

ABSTRACT

Multiview graph domain adaptation (MGDA) transfers knowledge from a labeled source graph to an unlabeled target graph, where nodes are connected through multiple relation views. We observe on real-world graphs that the source and target domains differ in their frequency composition and that this shift differs from one relation view to another. Existing MGDA methods align each view representation as a whole, leaving frequency-specific shifts uncorrected. We propose Frequency-Aware View Alignment (FAVA), which aligns the frequency composition of relation views across domains. For each relation view, FAVA learns three channels that separately capture low-frequency information, high-frequency variation, and full-pass node attributes, so that alignment can act on each frequency component individually. It then aligns these channels across all source–target view pairs, weighting each pair by its estimated shift to accommodate the view-dependent frequency shift. Experimental results on real-world graph benchmarks demonstrate the effectiveness of our method.

1 INTRODUCTION

In the era of large-scale graph data collection, real-world networks often integrate node attributes with diverse types of relations. To model these multi-relational structures, multiview graphs have emerged to represent each relation type as a distinct view over a shared node set. While multiview graphs are widely used in citation networks, social networks, and knowledge graphs (Wang et al., 2019; Park et al., 2020; Fu et al., 2020), performance on downstream tasks often suffers from a scarcity of node labels. Obtaining sufficient labels for graph learning tasks is difficult and costly, often leaving insufficient supervision for node classification.



(a) ACM1–ACM2 (PSP) (b) ACM1–ACM2 (PAP) (c) RU–US (PAP) (d) RU–US (PP)

Figure 1: View-dependent domain shifts in two groups of datasets. Frequency-related domain shifts exhibit distinct patterns across relation views. The complete figure is provided in Appendix B.

To address label scarcity, multiview graph domain adaptation (MGDA) has been proposed as a paradigm for transferring knowledge from a labeled source graph to an unlabeled target graph (Ben-David et al., 2010; Shi et al., 2025; Fang et al., 2026b). Existing unsupervised graph domain adaptation methods typically align source and target node representations through discrepancy minimization (Gretton et al., 2012; Chen et al., 2026a) or adversarial training (Ganin et al., 2016; Dai et al., 2022). Recent studies further model structural discrepancies across graph domains (Wu et al., 2023; Liu et al., 2024c). In this work, we compare frequency distributions between source and target domains in real-world multiview graphs and observe that frequency-related domain shifts vary across relation views, as shown in Figure 1. However, current alignment methods typically apply a single alignment objective to fused multiview representations

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

without explicitly modeling view-dependent domain shifts, which can reduce the effectiveness of cross-domain alignment, as illustrated in Figure 2(a).

Motivated by this observation, we propose Frequency-Aware View Alignment (FAVA), a novel framework that aligns the frequency composition of relation views across domains. As illustrated in Figure 2(b), FAVA explicitly aligns the frequency composition of individual relation views across domains rather than aligning fused multiview representations as a whole. For each relation view, FAVA constructs separate representations of low-frequency information, high-frequency variation, and full-pass node attributes. From these representations, each relation view's frequency composition is characterized using a channel-variance vector. FAVA then directly aligns the frequency compositions of corresponding source–target views and adaptively weights non-corresponding view pairs by their frequency-composition similarity. Finally, FAVA concatenates the channel representations from all relation views, combining information from different views and frequency channels for target-domain node classification.

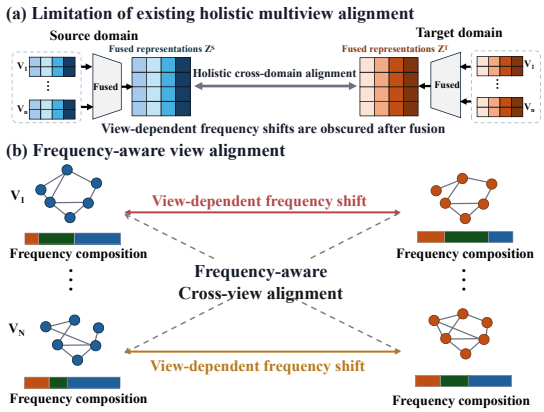


Figure 2: Existing alignment versus frequency-aware view alignment. (a) Existing alignment on fused representations can obscure view-dependent frequency shifts. (b) FAVA aligns frequency compositions across all source-target view pairs.

Our main contributions are summarized as follows:

- We compare source and target frequency distributions in real-world multiview graphs and observe that frequency-related domain shifts differ across relation views.
- We propose FAVA, a frequency-aware view alignment framework for MGDA that addresses view-dependent frequency shifts by aligning frequency compositions across all source-target view pairs.
- Comprehensive experiments on real-world graph benchmarks show that FAVA outperforms representative baselines in cross-domain node classification.

2 Related Work

2.1 Graph Domain Adaptation

Unsupervised graph domain adaptation transfers knowledge from a labeled graph to an unlabeled graph (Shi et al., 2025). ADAlign learns an adaptive distribution discrepancy using neural characteristic functions (Chen et al., 2026a). DiffGDA models structure-semantic evolution through continuous-time diffusion (Chen et al., 2026b). DisCo combines feature disentanglement with diffusion-guided label denoising (Zhang et al., 2026). SLOGAN combines sparse causal modeling with generative intervention (Luo et al., 2025).

Structural shifts are another focus of GDA (Liu et al., 2024b). A2GNN investigates cross-domain generalization through graph propagation (Liu et al., 2024a). StruRW addresses conditional structure shift through structural reweighting (Liu et al., 2023); PairAlign jointly considers conditional structure shift and label shift (Liu et al., 2024c). PSAHS progressively adjusts graph structure to address homophily shifts (Wen et al.). JHGDA aligns representations across graph hierarchies (Shi et al., 2023). MuDiS-GDA develops node- and subgraph-level adaptation strategies (Zhang & Lei, 2025).

GDA also considers different supervision settings. SGDA and GIFI address limited source labels (Qiao et al., 2023; 2024). PALA and ImGDA use prototype-based learning for class-imbalanced adaptation (Ma et al., 2025; 2026). Source-free methods adapt without access to source graphs (Zhang et al., 2024; Luo et al., 2024; 2026). Open-set adaptation considers target

classes absent from the source label space (Yin et al., 2024; Zhang & He, 2026). Multi-source adaptation draws on several source graphs (Lin et al., 2025; Shen et al., 2026). Beyond node classification, DREAM combines message passing and path aggregation in a dual variational framework for graph-level adaptation (Yin et al., 2025). GMM uses graph structure and Gaussian mixtures for visual domain adaptation (Wang et al., 2025).

In heterogeneous graphs, meta-paths encode relation semantics (Fu et al., 2020). HAN integrates meta-path representations through semantic-level attention (Wang et al., 2019). HeCo contrasts network-schema and meta-path views (Wang et al., 2021). For heterogeneous graph adaptation, MuSDAC learns in multiple semantic spaces (Yang et al., 2021). HGA aligns semantic-specific hierarchical information across domains (Zhuang et al., 2021). Multiview graph learning combines relation structures to learn complementary node representations (Park et al., 2020). SAGA dynamically selects relation-view and neighborhood-order combinations to guide source–target alignment (Fang et al., 2026b).

2.2 Graph Signal Modeling

Graph signal processing analyzes node-attribute variations over graph structures in the frequency domain (Shuman et al., 2013). APPNP uses personalized PageRank propagation (Gasteiger et al., 2018). GPR-GNN and BernNet learn filter coefficients to adjust frequency responses (Chien et al., 2020; He et al., 2021). FAGCN adaptively combines low- and high-frequency information (Bo et al., 2021). ACM-GCN combines low-pass, high-pass, and identity channels (Luan et al., 2022).

Graph signal modeling has also been applied to domain adaptation. SpecReg regularizes spectral properties to improve cross-domain transfer (You et al., 2023). SA-GDA introduces spectral augmentation (Pang et al., 2023), and DGSDA separates attribute alignment from spectral filter alignment (Yang et al., 2025). GAA examines node attribute shifts in cross-domain transfer (Fang et al., 2025a). HGDA addresses homophily discrepancies by separately aligning homophilic, heterophilic, and attribute representations (Fang et al., 2025b). RSGDA reconstructs homophilic and heterophilic graphs for cross-domain alignment (Fang et al., 2026a). For graph-level tasks, FracNet combines low-frequency and high-frequency decomposition with distribution matching and contrastive learning (Zhang et al., 2025). Although these methods have considered spectral information and different graph signals, they do not explicitly model view-dependent frequency shifts in MGDA.

3 Method

In this section, we present the formulation of FAVA for multiview graph domain adaptation. FAVA mainly consists of frequency-aware multichannel representation, cross-view frequency alignment, and node prediction. The overall architecture is shown in Figure 3. The following subsections describe these components and the optimization objective in detail.

3.1 Problem Definition

We consider unsupervised multiview graph domain adaptation for node classification. Given a labeled source multiview graph G^S and an unlabeled target multiview graph G^T, the two domains share a set of relation views $\mathcal{M}$. Each relation view $v \in \mathcal{M}$ corresponds to a relation type or a meta-path. The source graph G^S and target graph G^T are represented as

$$G^S = \left(\mathcal{N}^S, X^S, \{A_v^S\}_{v\in\mathcal{M}}, Y^S\right), \qquad G^T = \left(\mathcal{N}^T, X^T, \{A_v^T\}_{v\in\mathcal{M}}\right). \tag{1}$$

For either domain $u \in \{S, T\}$, $\mathcal{N}^u$ is the node set of size n^u, X^u is the node attribute matrix, and A_v^u is the symmetric adjacency matrix of view $v \in \mathcal{M}$. The source and target domains share the same label space with C classes, and $Y^S \in \{0,1\}^{n^S\times C}$ is the one-hot label matrix of the source nodes. Target node labels are unavailable during training.

The source and target domains may differ in their node-attribute and relation-structure distributions (Chen et al., 2026b). The goal of FAVA is to predict target-domain node labels using source-domain supervision and the unlabeled target graph.

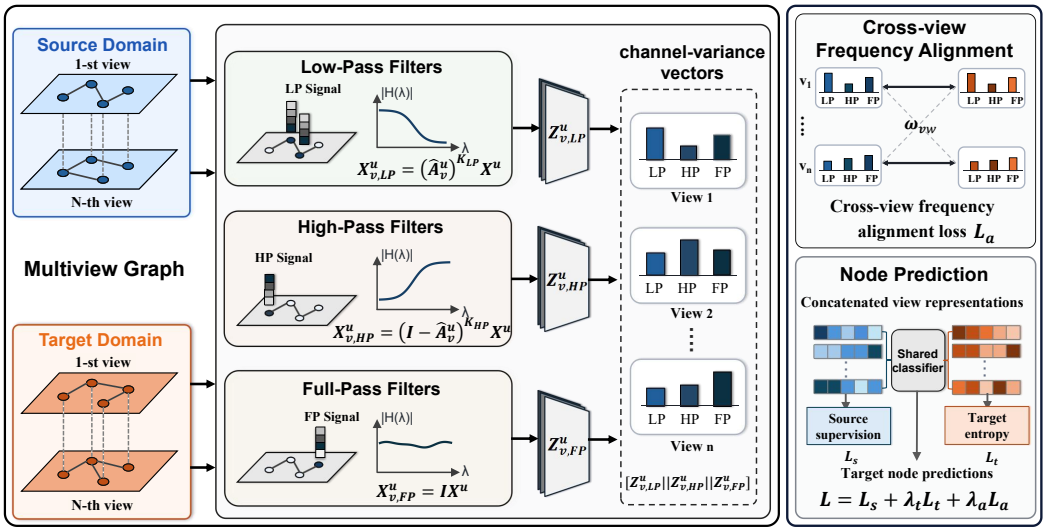


Figure 3: Overall architecture of FAVA. For each relation view, FAVA applies low-pass, high-pass, and full-pass graph filters and separately encodes the resulting signals into node representations. It summarizes their total variances as a normalized channel-variance vector that characterizes the view's frequency composition. Cross-view frequency alignment minimizes discrepancies between frequency compositions across all source-target view pairs, with adaptive weights for non-corresponding pairs. For node prediction, FAVA concatenates the channel representations across views and feeds them into a shared classifier trained with source supervision and target entropy.

3.2 Frequency-Aware Multichannel Representations

FAVA constructs frequency-aware representations separately for each relation view through graph filtering. For relation view v in domain u, we define the symmetrically normalized adjacency matrix with self-loops $\widehat{A}_v^u$ as (Kipf & Welling, 2016):

$$\widehat{A}_v^u = (D_v^u)^{-\frac{1}{2}} (A_v^u + I) (D_v^u)^{-\frac{1}{2}} \tag{2}$$

where D_v^u is the diagonal degree matrix of $A_v^u + I$ with I the identity matrix.

FAVA constructs low-pass (LP), high-pass (HP), and full-pass (FP) graph filters for each relation view. LP filter uses $\widehat{A}_v^u$ to aggregate attributes from neighboring nodes and capture low-frequency information. HP filter applies $I - \widehat{A}_v^u$ to calculate the difference between the input graph signal and its neighborhood aggregation, capturing high-frequency variation. FP filter applies the identity operator I to preserve information across the full frequency spectrum. The three filtered graph signals are given by:

$$X_{v,\mathrm{LP}}^u = \left(\widehat{A}_v^u\right)^{K_{\mathrm{LP}}} X^u, \qquad X_{v,\mathrm{HP}}^u = \left(I - \widehat{A}_v^u\right)^{K_{\mathrm{HP}}} X^u, \qquad X_{v,\mathrm{FP}}^u = IX^u \tag{3}$$

where K_{LP} and K_{HP} are the LP and HP filter orders, respectively. $X_{v,\mathrm{LP}}^u$, $X_{v,\mathrm{HP}}^u$, and $X_{v,\mathrm{FP}}^u$ denote the LP, HP, and FP filtered signals of view v in domain u.

We treat X^u as a graph signal defined on the nodes and λ as a graph frequency, corresponding to an eigenvalue of the normalized graph Laplacian $I - \widehat{A}_v^u$ (Shuman et al., 2013; He et al., 2021). The three filters have frequency responses $g_{\mathrm{LP}}(\lambda) = (1-\lambda)^{K_{\mathrm{LP}}}$, $g_{\mathrm{HP}}(\lambda) = \lambda^{K_{\mathrm{HP}}}$, and $g_{\mathrm{FP}}(\lambda) = 1$.

FAVA applies row-wise ℓ_2 normalization to each filtered graph signal to control the scale of the encoder inputs. Before encoding, we refer to the LP, HP, and FP representation pathways as three channels. Then three filtered graph signals are separately encoded into the LP, HP, and FP channel representations $Z_{v,\mathrm{LP}}^u$, $Z_{v,\mathrm{HP}}^u$, and $Z_{v,\mathrm{FP}}^u$:

$$Z_{v,\mathrm{LP}}^u = f_{\mathrm{LP}}\left(X_{v,\mathrm{LP}}^u; \theta_{\mathrm{LP}}\right), \quad Z_{v,\mathrm{HP}}^u = f_{\mathrm{HP}}\left(X_{v,\mathrm{HP}}^u; \theta_{\mathrm{HP}}\right), \quad Z_{v,\mathrm{FP}}^u = f_{\mathrm{FP}}\left(X_{v,\mathrm{FP}}^u; \theta_{\mathrm{FP}}\right) \tag{4}$$

where f_{LP}, f_{HP}, and f_{FP} are learnable channel encoders with parameters θ_{LP}, θ_{HP}, and θ_{FP}, respectively. For each channel, the encoder is shared by all relation views in both source and target

domains. The total variances of the channel representations are utilized to construct normalized channel-variance vectors for cross-view frequency alignment.

3.3 Cross-View Frequency Alignment

FAVA characterizes the frequency composition of each relation view using the relative total variances of the LP, HP, and FP channel representations. For a channel representation Z, the total variance $V(Z)$ is defined as

$$V(Z) = \frac{1}{n^u} \sum_{i=1}^{n^u} \|Z_{i,:} - \mu_Z\|_2^2 \tag{5}$$

where $Z_{i,:}$ denotes the representation of node i, $\mu_Z = \frac{1}{n^u} \sum_{i=1}^{n^u} Z_{i,:}$ is the mean representation across nodes, and $\| \cdot \|_2$ denotes the Euclidean norm. For relation view v in domain u, FAVA computes the total variances of the LP, HP, and FP channel representations, denoted by $s^u_{v,\mathrm{LP}}$, $s^u_{v,\mathrm{HP}}$, and $s^u_{v,\mathrm{FP}}$, respectively:

$$s^u_{v,\mathrm{LP}} = V\big(Z^u_{v,\mathrm{LP}}\big)\,, \qquad s^u_{v,\mathrm{HP}} = V\big(Z^u_{v,\mathrm{HP}}\big)\,, \qquad s^u_{v,\mathrm{FP}} = V\big(Z^u_{v,\mathrm{FP}}\big)\,. \tag{6}$$

The normalized channel-variance vector p^u_v is defined from the three channel variances as

$$p^u_v = \frac{\big[s^u_{v,\mathrm{LP}},\ s^u_{v,\mathrm{HP}},\ s^u_{v,\mathrm{FP}}\big]}{s^u_{v,\mathrm{LP}} + s^u_{v,\mathrm{HP}} + s^u_{v,\mathrm{FP}}}. \tag{7}$$

Each entry of p^u_v is the normalized variance of the corresponding channel. The three entries together characterize the frequency composition of view v. This normalization makes p^u_v reflect the relative variances among the three channels rather than the absolute variance values.

FAVA assigns a fixed alignment weight to corresponding source-target views and adaptive weights to non-corresponding view pairs based on frequency-composition similarity. For source view v and target view w, the frequency-composition similarity o_{vw} is calculated by summing the channel-wise overlap between the two normalized channel-variance vectors:

$$o_{vw} = \min\big(p^S_{v,\mathrm{LP}}, p^T_{w,\mathrm{LP}}\big) + \min\big(p^S_{v,\mathrm{HP}}, p^T_{w,\mathrm{HP}}\big) + \min\big(p^S_{v,\mathrm{FP}}, p^T_{w,\mathrm{FP}}\big)\,. \tag{8}$$

where $\min$ denotes the minimum operator. A fixed alignment weight of 1 is assigned to the corresponding views and an adaptive weight to each non-corresponding view pair based on frequency-composition similarity:

$$\omega_{vw} = \begin{cases} 1, & v = w, \\ \dfrac{o_{vw}}{2(M-1)}, & v \neq w \end{cases} \tag{9}$$

where M is the number of relation views, and ω_{vw} is the alignment weight between source view v and target view w. Using these alignment weights, the cross-view frequency alignment loss $\mathcal{L}_{\mathrm{a}}$ is defined as the weighted average discrepancy between normalized channel-variance vectors over all source-target view pairs:

$$\mathcal{L}_{\mathrm{a}} = \frac{\displaystyle\sum_{v\in\mathcal{M}} \sum_{w\in\mathcal{M}} \omega_{vw} \left\|p^S_v - p^T_w\right\|_2^2}{\displaystyle\sum_{v\in\mathcal{M}} \sum_{w\in\mathcal{M}} \omega_{vw}}. \tag{10}$$

At each forward pass, FAVA derives the adaptive weights from the current channel-variance vectors. During the backward pass, the adaptive weights are treated as constants, while gradients from $\mathcal{L}_{\mathrm{a}}$ propagate through p^S_v and p^T_w to update the channel encoders.

3.4 Node Prediction and Optimization

For node prediction, FAVA first concatenates the three channel representations within each relation view and then combines the resulting view representations into Z^u:

$$Z^u = \Big\|_{v\in\mathcal{M}} \big(Z^u_{v,\mathrm{LP}} \| Z^u_{v,\mathrm{HP}} \| Z^u_{v,\mathrm{FP}}\big) \tag{11}$$

where $\|$ denotes concatenation along the feature dimension.

A linear classifier shared across the source and target domains is utilized to obtain the class-probability matrix Q^u:

$$Q^u = \mathrm{softmax}\left(Z^u W_c + \mathbf{1}_{n^u} b_c^\top\right) \tag{12}$$

where W_c is the classifier weights, b_c is the bias, and $\mathbf{1}_{n^u}$ is an all-ones column vector of length n^u. Softmax is applied along the class dimension, and $Q^u_{i,c}$ denotes the predicted probability that node i belongs to class c.

During the training, source-domain cross-entropy is minimized to learn from labeled nodes and target-domain conditional entropy to encourage confident predictions (Grandvalet & Bengio, 2004):

$$\mathcal{L}_s = -\frac{1}{n^S} \sum_{i=1}^{n^S} \sum_{c=1}^{C} Y^S_{i,c} \log Q^S_{i,c}, \tag{13}$$

$$\mathcal{L}_t = -\frac{1}{n^T} \sum_{i=1}^{n^T} \sum_{c=1}^{C} Q^T_{i,c} \log Q^T_{i,c} \tag{14}$$

where $Y^S_{i,c}$ is the one-hot label entry, equal to 1 if source node i belongs to class c and 0 otherwise.

Combining the prediction losses with the cross-view frequency alignment loss gives the overall training objective:

$$\mathcal{L} = \mathcal{L}_s + \lambda_t \mathcal{L}_t + \lambda_a \mathcal{L}_a \tag{15}$$

where λ_t and λ_a control the contributions of target conditional entropy and cross-view frequency alignment, respectively. The channel encoders and shared classifier are optimized by this objective. Finally, FAVA uses the trained model to predict node labels on the target graph.

4 Experiments

4.1 Experimental Setup

We evaluate FAVA on eight real-world multiview graphs across 12 cross-domain transfer tasks. The detailed experimental settings are provided in Appendix A.

Datasets. The eight graphs include two subgraphs from ACM and six subgraphs from MAG where nodes represent papers, paper attributes serve as node features, and different relations define multiple graph views. ACM1 and ACM2 divided from ACM contain two meta-path-based views: Paper-Author-Paper (PAP) and Paper-Subject-Paper (PSP). Their node labels correspond to three research areas: database, wireless communication, and data mining. MAG is derived from Microsoft Academic Graph (Wang et al., 2020) and contains six country-specific subgraphs: CN, US, JP, FR, DE, and RU. Each subgraph contains two views: Paper-Author-Paper (PAP) and Paper-Paper (PP). The node classes correspond to the ten most frequent research topics in the benchmark. We evaluate transfer in both directions between ACM1 and ACM2 and on ten directed transfer tasks among the MAG subgraphs. The transfer directions are listed in Tables 2 and 3.

Table 1: Statistics of the ACM and MAG datasets

Graph	#Nodes	Relation Views	#Edges / View	Feature Dimension	#Classes
ACM1	3,025	PAP / PSP	26,256 / 2,207,736	512	3
ACM2	4,019	PAP / PSP	57,853 / 4,338,213	512	3
CN	25,467	PAP / PP	157,303 / 43,683	128	10
US	55,392	PAP / PP	12,332,742 / 289,833	128	10
JP	12,651	PAP / PP	41,165 / 20,747	128	10
FR	12,052	PAP / PP	51,739 / 30,915	128	10
DE	17,919	PAP / PP	77,100 / 45,528	128	10
RU	3,690	PAP / PP	3,315 / 3,539	128	10

Baselines. We compare our proposed FAVA with graph domain adaptation methods and non-adaptive methods. The non-adaptive baselines are GCN (Kipf & Welling, 2016) and GAT (Veličković et al., 2017), which use graph convolution and attention-based neighborhood aggregation respectively. Among the graph domain adaptation methods, AdaGCN (Dai et al., 2022) and ACDNE (Shen et al., 2020) use adversarial learning to align node representations across domains. UDA-GCN (Wu et al., 2020) jointly encodes local and global structure, while ASN (Zhang et al., 2021) separates shared and private representations. GRADE-N (Wu et al., 2023) reduces graph subtree discrepancies, and PairAlign (PA) (Liu et al., 2024c) addresses conditional structure shift and label shift. SpecReg (You et al., 2023) uses spectral regularization, whereas HGDA (Fang et al., 2025b) separately aligns homophilic, heterophilic, and attribute representations. GraphATA (Zhang & He, 2025) aggregates source models at the node level for source-free adaptation. SAGA (Fang et al., 2026b) dynamically selects combinations of relation views and neighborhood orders to guide alignment. To apply single-view baselines to multiview graphs, we calculate node representations separately for each relation view and average them across views before classification. The same procedure is applied to both source and target graphs. We evaluate node classification using Accuracy and Macro-F1. Accuracy measures the overall proportion of correctly classified nodes, while Macro-F1 averages F1 scores across classes with equal weight. Higher values indicate better performance.

Table 2: Experiment on ACM.

Method	ACM1→ACM2		ACM2→ACM1	
	Acc.	Macro-F1	Acc.	Macro-F1
GCN	0.367	0.253	0.359	0.236
GAT	0.373	0.319	0.363	0.241
AdaGCN	0.431	0.272	0.386	0.362
UDA-GCN	0.452	0.283	0.409	0.347
GRADE-N	0.469	0.291	0.373	0.343
ACDNE	0.457	0.261	0.392	0.363
ASN	0.436	0.321	0.437	0.327
SpecReg	0.493	0.308	0.431	0.381
PA	0.506	0.333	0.452	0.377
GraphATA	0.501	0.322	0.401	0.395
HGDA	0.511	0.311	0.441	0.414
SAGA	0.523	0.351	0.454	0.444
FAVA	**0.679**	**0.608**	**0.644**	**0.591**

Table 3: Cross-domain node classification results on MAG

Method	Metric	CN→US	US→CN	JP→CN	CN→JP	DE→FR	FR→DE	RU→US	US→RU	DE→CN	CN→DE
GCN	Acc.	0.351	0.263	0.327	0.335	0.319	0.302	0.301	0.317	0.281	0.291
	Macro-F1	0.217	0.218	0.221	0.201	0.203	0.167	0.233	0.320	0.167	0.223
GAT	Acc.	0.325	0.316	0.361	0.309	0.324	0.253	0.265	0.339	0.281	0.316
	Macro-F1	0.218	0.320	0.253	0.237	0.212	0.249	0.221	0.323	0.249	0.209
AdaGCN	Acc.	0.418	0.483	0.389	0.425	0.431	0.314	0.370	0.435	0.359	0.363
	Macro-F1	0.231	0.362	0.257	0.244	0.242	0.285	0.242	0.369	0.265	0.212
UDA-GCN	Acc.	0.425	0.453	0.451	0.414	0.352	0.364	0.359	0.421	0.356	0.369
	Macro-F1	0.243	0.341	0.271	0.275	0.274	0.262	0.239	0.361	0.296	0.253
GRADE-N	Acc.	0.421	0.467	0.376	0.404	0.455	0.375	0.353	0.429	0.358	0.362
	Macro-F1	0.241	0.352	0.305	0.292	0.286	0.287	0.245	0.347	0.299	0.252
ACDNE	Acc.	0.410	0.458	0.422	0.436	0.463	0.359	0.351	0.434	0.360	0.378
	Macro-F1	0.225	0.352	0.335	0.275	0.279	0.292	0.231	0.388	0.265	0.291
ASN	Acc.	0.416	0.431	0.335	0.432	0.433	0.368	0.289	0.435	0.334	0.372
	Macro-F1	0.232	0.387	0.219	0.251	0.248	0.297	0.216	0.226	0.258	0.270
SpecReg	Acc.	0.425	0.501	0.363	0.419	0.424	0.387	0.328	0.429	0.330	0.371
	Macro-F1	0.247	0.380	0.273	0.263	0.236	0.282	0.213	0.371	0.279	0.303
PA	Acc.	0.402	0.529	0.351	0.323	0.416	0.357	0.306	0.317	0.336	0.357
	Macro-F1	0.236	0.331	0.295	0.243	0.229	0.299	0.235	0.367	0.286	0.263
GraphATA	Acc.	0.407	0.518	0.382	0.341	0.327	0.414	0.351	0.341	0.341	0.321
	Macro-F1	0.241	0.339	0.287	0.252	0.291	0.289	0.234	0.351	0.274	0.254
HGDA	Acc.	0.417	0.529	0.392	0.343	0.391	0.421	0.361	0.337	0.361	0.317
	Macro-F1	0.239	0.333	0.321	0.251	0.284	0.279	0.212	0.344	0.271	0.249
SAGA	Acc.	0.553	0.535	0.512	0.557	0.529	0.515	0.377	0.441	0.425	0.381
	Macro-F1	0.341	0.426	0.372	0.415	0.426	0.401	0.378	0.397	0.320	0.350
FAVA	Acc.	**0.611**	**0.545**	**0.553**	**0.598**	**0.563**	**0.560**	**0.476**	**0.492**	**0.485**	**0.448**
	Macro-F1	**0.443**	**0.438**	**0.433**	**0.466**	**0.480**	**0.483**	**0.403**	**0.447**	**0.428**	**0.432**

4.2 Performance Comparison

Tables 2 and 3 summarize the results on all 12 transfer tasks. FAVA achieves the highest Accuracy and Macro-F1 on every task. Among the compared baselines, SAGA shows the strongest overall performance and is therefore used as the main reference for comparison. On ACM, FAVA obtains average Accuracy and Macro-F1 scores of 66.2% and 60.0%, compared with 48.9% and 39.8% for SAGA, with absolute improvements of 17.3 and 20.2 percentage points, respectively. On MAG, the corresponding averages are 53.3% and 44.5%, exceeding SAGA by 5.1 and 6.3 percentage points.

Non-adaptive methods generally perform worse than domain adaptation methods, indicating the importance of reducing source-target distribution discrepancies. Among the adaptation baselines, SAGA is the closest to FAVA because it explicitly considers multiple relation views during alignment. Nevertheless, FAVA outperforms SAGA on all 12 transfer tasks. The larger gains on ACM are consistent with the stronger differences between its PAP and PSP views, while the gains on MAG remain consistent across all ten transfer directions. This comparison suggests that modeling relation views alone is not sufficient when frequency shifts vary across views.

4.3 Ablation Study

We evaluate the contributions of frequency alignment, cross-view alignment, and the three representation channels of LP, HP and FP on datasets. We compare the following variants:

- **w/o Frequency Alignment**: Set $\lambda_a = 0$ to remove the cross-view frequency alignment objective, while retaining all three channels and all relation views for node classification.
- **w/o Cross-View Alignment**: Set $\omega_{vw} = 0$ for $v \neq w$, retaining alignment only between corresponding source-target views. All relation views are still used for node classification.
- **LP Only, HP Only, and FP Only**: Retain only the low-pass, high-pass, or full-pass channel, respectively, while keeping all relation views for node classification.
- **FAVA (Full)**: Use all three channels and cross-view frequency alignment across all source-target view pairs, with adaptive weights for non-corresponding pairs.

Figure 4 shows that Full FAVA performs better than all variants in both ACM transfer directions. Removing either frequency alignment or non-corresponding view alignment reduces classification performance, showing that both components contribute to FAVA. The single-channel variants also perform consistently worse than FAVA, indicating that low-frequency information, high-frequency variation, and full-pass node attributes provide complementary information for cross-domain transfer. The same pattern is observed across all ten MAG transfer tasks. The complete ablation results on MAG are reported in Appendix E.

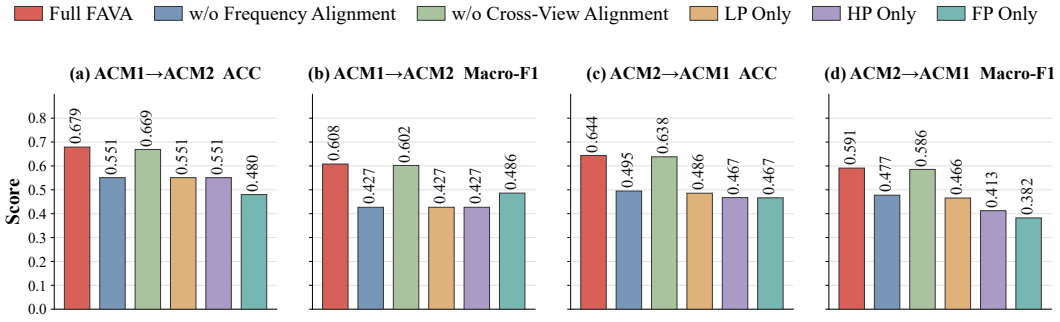
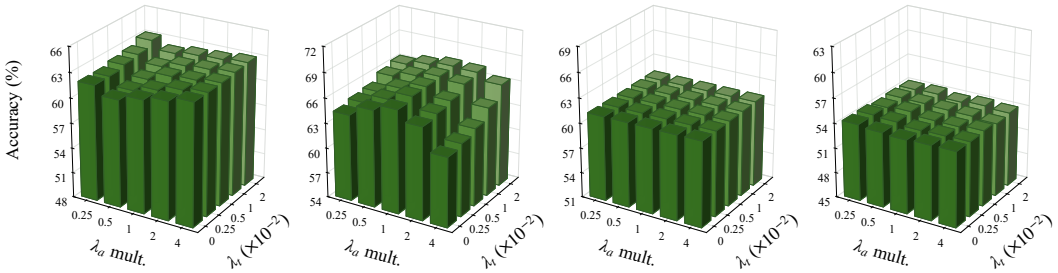


Figure 4: Ablation results.

4.4 Hyperparameter Sensitivity Analysis

We examine the sensitivity of FAVA to λ_a and λ_t on ACM1→ACM2, ACM2→ACM1, CN→US, and US→CN, as shown in Figure 5. FAVA remains highly stable on two MAG tasks across the

tested parameter range, indicating that its performance does not depend on a narrow choice of hyperparameters. The ACM tasks are more sensitive, but strong results are still obtained over several neighboring settings, especially with moderate alignment weights. Overall, FAVA maintains a broad effective parameter range across the two benchmarks. The complete sensitivity results for all 12 transfer tasks are provided in Appendix D.

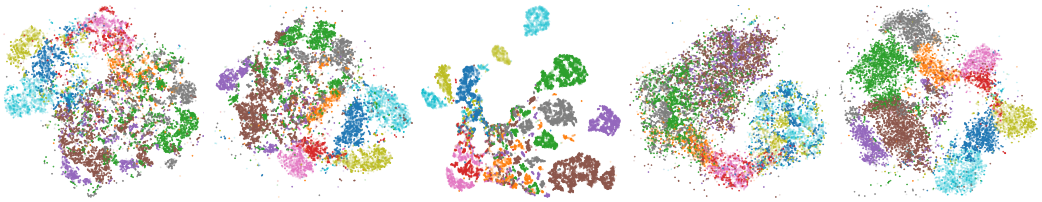


(a) ACM1→ACM2 (b) ACM2→ACM1 (c) CN→US (d) US→CN

Figure 5: Sensitivity to λ_{a} and λ_{t}.

4.5 Node Embedding Visualization

In this section, we visualize the node embeddings generated by representative GDA methods on the DE→CN task. As shown in Figure 6, FAVA produces clearer class-wise clusters than the compared methods. Each color represents a different node class, while the lighter and darker shades within each class denote nodes from the source domain and target domain, respectively. First, nodes from different classes are more distinctly separated, indicating better class discrimination in the embedding space. Second, nodes belonging to the same class are generally more compact, source and target nodes within the same class show greater overlap, and several baselines exhibit stronger class mixing or fragmented clusters. These observations suggest that FAVA learns more discriminative and transferable node representations for cross-domain classification. The complete visualization results for all ten MAG transfer tasks are provided in Appendix G.



(a) UDA-GCN (b) ASN (c) PairAlign (d) SAGA (e) FAVA

Figure 6: Visualizing source and target node embeddings via T-SNE. Each color represents a class, while light and dark shades of the same color correspond to the same class nodes from the source and target domains, respectively.

5 Conclusion

Our analysis of real-world graphs shows that frequency differences between source and target domains vary across relation views. Motivated by this observation, we propose Frequency-Aware View Alignment (FAVA) for multiview graph domain adaptation. FAVA separately encodes low-frequency information, high-frequency variation, and full-pass node attributes, and characterizes each view's frequency composition with a normalized channel-variance vector. It aligns frequency compositions across all source-target view pairs, with non-corresponding pairs adaptively weighted according to frequency-composition similarity. Experiments on eight real-world graphs show that our proposed FAVA outperforms representative baselines in both Accuracy and Macro-F1 on all 12 transfer tasks. Future work will extend FAVA to source and target graphs with partially overlapping or unmatched relation-view sets.

6 AI USE STATEMENT

In this work, we used generative AI tools solely for language polishing. LLMs were not involved in conceptual design, theoretical formulation, experimental implementation, or result analysis. We have reviewed all AI-assisted revisions for accuracy, clarity, and consistency with the original technical content. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 3950–3957, 2021.

Wei Chen, Xingyu Guo, Shuang Li, Zhao Zhang, Yan Zhong, Fuzhen Zhuang, and deqing wang. Learning adaptive distribution alignment with neural characteristic function for graph domain adaptation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 17271–17294, 2026a.

Wei Chen, Xingyu Guo, Shuang Li, Yan Zhong, Zhao Zhang, Fuzhen Zhuang, Hongrui Liu, Libang Zhang, Guo Ye, and Huimei He. Learning structure-semantic evolution trajectories for graph domain adaptation. *International Conference on Learning Representations*, 2026b.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*, 2020.

Quanyu Dai, Xiao-Ming Wu, Jiaren Xiao, Xiao Shen, and Dan Wang. Graph transfer learning via adversarial domain adaptation with graph convolution. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4908–4922, 2022.

Ruiyi Fang, Bingheng Li, Qiuhao Zeng, Nima Hosseini Dashtbayaz, Ruizhi Pu, Charles Ling, Boyu Wang, et al. On the benefits of attribute-driven graph domain adaptation. In *International conference on learning representations*, volume 2025, pp. 30762–30782, 2025a.

Ruiyi Fang, Bingheng Li, Jingyu Zhao, Ruizhi Pu, Qiuhao Zeng, Gezheng Xu, Charles Ling, and Boyu Wang. Homophily enhanced graph domain adaptation. *arXiv preprint arXiv:2505.20089*, 2025b.

Ruiyi Fang, Shuo Wang, Ruizhi Pu, Qiuhao Zeng, Hao Zheng, Ziyan Wang, Jiale Cai, Zhimin Mei, Song Tang, Charles Ling, et al. Graph domain adaptation via homophily-agnostic reconstructing structure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 21047–21055, 2026a.

Ruiyi Fang, Jingyu Zhao, Shuo Wang, Ruizhi Pu, Bingheng Li, Jiale Cai, Zhihao Li, Zihao Jing, Jian Zhu, Song Tang, et al. Saga: Structural aggregation guided alignment with dynamic view and neighborhood order selection for multiview graph domain adaptation. In *International Conference on Learning Representations*, volume 2026, pp. 95585–95607, 2026b.

Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding. In *Proceedings of the web conference 2020*, pp. 2331–2341, 2020.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. *arXiv preprint arXiv:1810.05997*, 2018.

Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The journal of machine learning research*, 13:723–773, 2012.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in neural information processing systems*, 34:14239–14251, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Chuanyun Lin, Xi Zhou, and Xiao Shen. Hierarchical graph contrastive domain adaptation for multi-source cross-network node classification. *Expert Systems with Applications*, 284:127900, 2025.

Meihan Liu, Zeyu Fang, Zhen Zhang, Ming Gu, Sheng Zhou, Xin Wang, and Jiajun Bu. Rethinking propagation for unsupervised graph domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13963–13971, 2024a.

Meihan Liu, Zhen Zhang, Jiachen Tang, Jiajun Bu, Bingsheng He, and Sheng Zhou. Revisiting, benchmarking and understanding unsupervised graph domain adaptation. *Advances in neural information processing systems*, 37:89408–89436, 2024b.

Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural re-weighting improves graph domain adaptation. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21778–21793. PMLR, 23–29 Jul 2023.

Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 32552–32575. PMLR, 21–27 Jul 2024c.

Sitao Luan, Chenqing Hua, Qincheng Lu, Jiaqi Zhu, Mingde Zhao, Shuyuan Zhang, Xiao-Wen Chang, and Doina Precup. Revisiting heterophily for graph neural networks. *Advances in neural information processing systems*, 35:1362–1375, 2022.

Junyu Luo, Zhiping Xiao, Yifan Wang, Xiao Luo, Jingyang Yuan, Wei Ju, Langechuan Liu, and Ming Zhang. Rank and align: Towards effective source-free graph domain adaptation. In Kate Larson (ed.), *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 4706–4714. International Joint Conferences on Artificial Intelligence Organization, 8 2024. doi: 10.24963/ijcai.2024/520.

Junyu Luo, Yuhao Tang, Yiwei Fu, Xiao Luo, Zhizhuo KOU, Zhiping Xiao, Wei Ju, Wentao Zhang, and Ming Zhang. Sparse causal discovery with generative intervention for unsupervised graph domain adaptation. In *Forty-second International Conference on Machine Learning*, 2025.

Junyu Luo, Haoyu Tao, Xiao Luo, Yusheng Zhao, Zhiping Xiao, Dailan He, Wei Ju, Chong Chen, Xian-Sheng Hua, and Ming Zhang. Robust cross supervision with target mining for source-free graph domain adaptation. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

Xin Ma, Yifan Wang, Siyu Yi, Wei Ju, Bei Wu, Ziyue Qiao, Chenwei Tang, and Jiancheng Lv. Pala: Class-imbalanced graph domain adaptation via prototype-anchored learning and alignment. In *IJCAI*, pp. 3198–3207, 2025.

Xin Ma, Yifan Wang, Siyu Yi, Wei Ju, Junyu Luo, Yusheng Zhao, Xiao Luo, and Jiancheng Lv. Dual prototype-enhanced contrastive framework for class-imbalanced graph domain adaptation. *Advances in Neural Information Processing Systems*, 38:48456–48482, 2026.

Jinhui Pang, Zixuan Wang, Jiliang Tang, Mingyan Xiao, and Nan Yin. Sa-gda: Spectral augmentation for graph domain adaptation. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 309–318, 2023.

Chanyoung Park, Donghyun Kim, Jiawei Han, and Hwanjo Yu. Unsupervised attributed multiplex network embedding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5371–5378, 2020.

Ziyue Qiao, Xiao Luo, Meng Xiao, Hao Dong, Yuanchun Zhou, and Hui Xiong. Semi-supervised domain adaptation in graph transfer learning. In Edith Elkind (ed.), *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pp. 2279–2287. International Joint Conferences on Artificial Intelligence Organization, 8 2023. Main Track.

Ziyue Qiao, Meng Xiao, Weiyu Guo, Xiao Luo, and Hui Xiong. Information filtering and interpolating for semi-supervised graph domain adaptation. *Pattern Recognition*, 153:110498, 2024.

Xiao Shen, Quanyu Dai, Fu-lai Chung, Wei Lu, and Kup-Sze Choi. Adversarial deep network embedding for cross-network node classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 2991–2999, 2020.

Xiao Shen, Chuanyun Lin, Zhihao Chen, Junwei Gong, and Xi Zhou. Prototypical multi-source graph contrastive domain adaptation. *Pattern Recognition*, pp. 114312, 2026.

Bo-Shen Shi, Yong-Qing Wang, Fang-Da Guo, Bing-Bing Xu, Hua-Wei Shen, and Xue-Qi Cheng. Domain adaptation for graph representation learning: Challenges, progress, and prospects. *Journal of Computer Science and Technology*, 40(2):283–300, 2025.

Boshen Shi, Yongqing Wang, Fangda Guo, Jiangli Shao, Huawei Shen, and Xueqi Cheng. Improving graph domain adaptation with network hierarchy. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pp. 2249–2258, 2023.

David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

Kuansan Wang, Zhihong Shen, Chiyuan Huang, Chieh-Han Wu, Yuxiao Dong, and Anshul Kanakia. Microsoft academic graph: When experts are not enough. *Quantitative Science Studies*, 1(1): 396–413, 2020.

Mengzhu Wang, Wenhao Ren, Yu Zhang, Yanlong Fan, Dianxi Shi, Luoxi Jing, and Nan Yin. Gaussian mixture model for graph domain adaptation. In *IJCAI*, pp. 1963–1972, 2025.

Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The world wide web conference*, pp. 2022–2032, 2019.

Xiao Wang, Nian Liu, Hui Han, and Chuan Shi. Self-supervised heterogeneous graph neural network with co-contrastive learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 1726–1736, 2021.

Hongwei Wen, Can Zhang, Haoyu He, Hanyuan Hang, and Minglong Lei. Progressive graph structure adjustment for homophily shift adaptation. In *Forty-third International Conference on Machine Learning*.

Jun Wu, Jingrui He, and Elizabeth Ainsworth. Non-iid transfer learning on graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 10342–10350, 2023.

Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. Unsupervised domain adaptive graph convolutional networks. In *Proceedings of the web conference 2020*, pp. 1457–1467, 2020.

Liang Yang, Xin Chen, Jiaming Zhuo, Di Jin, Chuan Wang, Xiaochun Cao, Zhen Wang, and Yuanfang Guo. Disentangled graph spectral domain adaptation. In *Forty-second International Conference on Machine Learning*, 2025.

Shuwen Yang, Guojie Song, Yilun Jin, and Lun Du. Domain adaptive classification on heterogeneous information networks. In *Proceedings of the Twenty-Ninth international conference on international joint conferences on artificial intelligence*, pp. 1410–1416, 2021.

Nan Yin, Mengzhu Wang, Zhenghan Chen, Li Shen, Huan Xiong, Bin Gu, and Xiao Luo. Dream: Dual structured exploration with mixup for open-set graph domain adaption. In *International conference on learning representations*, volume 2024, pp. 19776–19797, 2024.

Nan Yin, Li Shen, Mengzhu Wang, Xinwang Liu, Chong Chen, and Xian-Sheng Hua. Dream: a dual variational framework for unsupervised graph domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. Graph domain adaptation via theory-grounded spectral regularization. In *International Conference on Learning Representations*, volume 2023, 2023.

Can Zhang and Minglong Lei. Mudis-gda: Multiscale discriminative graph domain adaptation. *Neural Networks*, pp. 108412, 2025.

Haodong Zhang, Tao Ren, Changhu Wang, Yifan Wang, Wei Ju, Huaizhi Tang, Junyu Luo, Zimo Wang, Ziyue Qiao, Xian-Sheng Hua, et al. Disco: Diffusion-guided unbiased discriminative learning for unsupervised graph domain adaptation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 1892–1903, 2026.

Haoyu Zhang, Yuxuan Cheng, Wenqi Fan, Yulong Chen, and Yifan Zhang. Rethinking graph domain adaptation: A spectral contrastive perspective. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 448–464. Springer, 2025.

Xiaowen Zhang, Yuntao Du, Rongbiao Xie, and Chongjun Wang. Adversarial separation network for cross-network node classification. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pp. 2618–2626, 2021.

Zhen Zhang and Bingsheng He. Aggregate to adapt: Node-centric aggregation for multi-source-free graph domain adaptation. In *Proceedings of the ACM on Web Conference 2025*, pp. 4420–4431, 2025.

Zhen Zhang and Bingsheng He. Towards unsupervised open-set graph domain adaptation via dual reprogramming. *Advances in Neural Information Processing Systems*, 38:53320–53343, 2026.

Zhen Zhang, Meihan Liu, Anhui Wang, Hongyang Chen, Zhao Li, Jiajun Bu, and Bingsheng He. Collaborate to adapt: Source-free graph domain adaptation via bi-directional adaptation. In *Proceedings of the ACM Web Conference 2024*, pp. 664–675, 2024.

YuanXin Zhuang, Chuan Shi, Cheng Yang, Fuzhen Zhuang, and Yangqiu Song. Semantic-specific hierarchical alignment network for heterogeneous graph adaptation. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 335–350. Springer, 2021.

A Experimental Setup

We implement FAVA in PyTorch and conduct experiments on a server equipped with two Intel Xeon Platinum 8352V CPUs (72 cores and 144 threads in total) and two NVIDIA RTX A6000 GPUs, each with 48 GB of memory. Each channel encoder has two layers, with a hidden dimension of 256 and an output dimension of 128. FAVA is trained using the Adam optimizer (Kingma & Ba, 2014) with cosine learning-rate scheduling and gradient clipping at a threshold of 1. The LP and HP filters use two and one propagation steps, respectively, while the FP filter uses the identity operator. Across transfer tasks, the learning rate ranges from 2×10^{-4} to 10^{-2}, with weight decay selected from $\{10^{-5}, 10^{-4}, 10^{-3}\}$ and batch size from $\{500, 1000, 2000, 5000\}$. The dropout rate is selected from $\{0, 0.1, 0.3\}$, and training runs for 300 or 400 epochs. We repeat each experiment with five random seeds and report Accuracy and Macro-F1 for target-domain node classification.

Graph filtering is precomputed on the full graphs, while training and alignment use node mini-batches. Channel means and variances are computed within each domain's current batch using its actual node count, and are used to obtain the channel compositions and adaptive alignment weights.

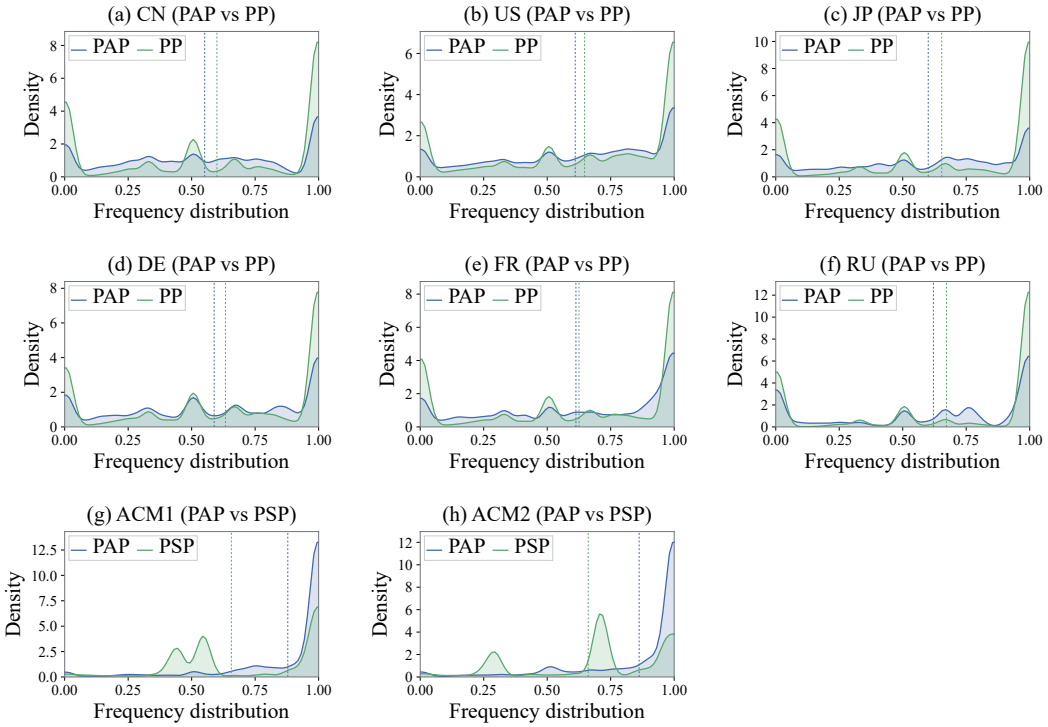


Figure 1: Frequency distributions of different relation views within each domain.

B Empirical Study of View-Dependent Frequency Distributions

To examine view-dependent frequency differences in multiview graphs, we analyze the six unique graph pairs underlying the 12 directed transfer tasks, together with all eight individual graphs. We consider two aspects: cross-domain differences under the same relation view and within-domain differences across relation views.

We regard node labels as a graph signal. For node i in domain u under relation view v, the same-label neighbor ratio is defined as

$$r_{v,i}^{u} = \frac{1}{d_{v,i}^{u}} \sum_{j \in \Gamma_{v}^{u}(i)} \mathbb{I}\left(y_{i}^{u} = y_{j}^{u}\right)$$

where $\Gamma_v^u(i)$ and $d_{v,i}^u$ denote the neighborhood and degree of node i in relation view v, respectively. This statistic measures the local smoothness of the label signal. A larger value indicates less label variation between adjacent nodes, whereas a smaller value indicates stronger local variation. Since smoother graph signals are associated with lower frequencies and stronger local variation with higher frequencies (Shuman et al., 2013), we use the empirical density of $r_{v,i}^u$ over all nodes to describe the frequency-related distribution under each relation view. The solid curves show the empirical density, and the dashed lines indicate the corresponding means. Target-domain labels are used only for the analysis in this section and are not involved in FAVA training.

Figure 1 compares the frequency-related distributions of different relation views within each domain. Across the six MAG graphs, the PAP and PP views show systematic differences in both distribution shape and mean. The PP view is generally more concentrated at larger same-label neighbor ratios, whereas the PAP view spreads more broadly over the middle-to-high range. The contrast is even more pronounced on ACM1 and ACM2: the PAP distribution is strongly concentrated near high ratios, while the PSP distribution places substantially more mass at intermediate values. These differences indicate that relation views within the same graph can exhibit distinct local smoothness and frequency characteristics.

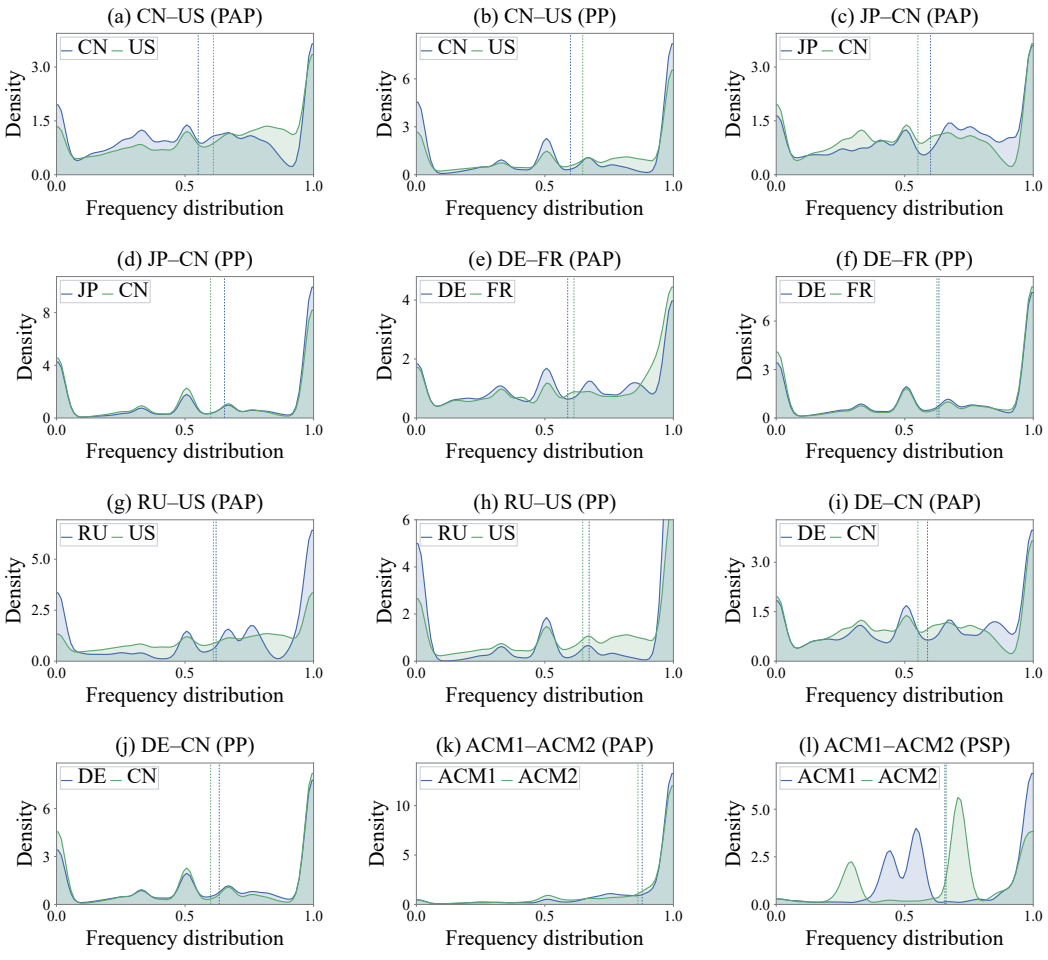


Figure 2: Cross-domain frequency distributions under different relation views.

Figure 2 compares the distributions of each graph pair under the same relation view. Since reversing the source and target domains does not change the two distributions being compared, each graph pair is shown only once. Across the six graph pairs, the two domains differ in both distribution shape and mean, but the pattern of this difference depends on the relation view. For the MAG pairs, the source-target separation under PAP differs from that under PP, with some pairs showing closer overlap in one view and a clearer shift in the other. A similar contrast appears for

ACM1-ACM2, where PAP and PSP exhibit markedly different source-target distribution patterns. Thus, the cross-domain frequency-related shift cannot be described by a single view-independent pattern.

Taken together, the cross-domain and within-domain comparisons show that frequency distributions differ both across domains and across relation views. Frequency differences in multiview graphs are therefore view-dependent, providing empirical support for modeling cross-domain frequency differences according to relation views in FAVA.

C Model Efficiency Analysis

We analyze FAVA's computational complexity and practical efficiency. Let $M = |\mathcal{M}|$ denote the number of relation views, $N = n^S + n^T$ the total number of nodes, and d an upper bound on input and encoder dimensions. With fixed filter orders, encoder depth, and number of classes, dense graph filtering costs $O(MN^2d)$ once before training, while each training iteration costs $O(MNd^2 + M^2)$. Cross-view alignment contributes $O(M^2)$ by comparing three-dimensional channel-variance vectors rather than source-target node pairs. The space complexity is $O(MN^2+MNd+d^2+M^2)$, including dense adjacency storage. In practice, FAVA uses sparse adjacency matrices. We compare its training time and GPU memory usage on 4 transfer tasks.

Table 1: Training time and peak GPU memory normalized to FAVA. Lower values indicate better efficiency.

Transfer Task	Method	Training Time	GPU Memory
CN→US	FAVA	**1.000**	**1.000**
	SAGA	1.885	2.966
	UDAGCN	1.370	5.414
	GraphATA	2.199	21.751
US→CN	FAVA	1.000	**1.000**
	SAGA	1.872	3.305
	UDAGCN	1.361	5.934
	GraphATA	**0.765**	5.190
ACM1→ACM2	FAVA	1.000	**1.000**
	SAGA	0.913	1.447
	UDAGCN	**0.798**	1.917
	GraphATA	0.858	4.461
ACM2→ACM1	FAVA	1.000	**1.000**
	SAGA	0.884	1.474
	UDAGCN	**0.787**	1.902
	GraphATA	0.846	8.533

Table 1 compares training time and peak GPU memory, with FAVA normalized to 1 for each task. On CN→US and US→CN, FAVA reduces training time by 46.6%–47.0% and peak GPU memory by 66.3%–69.7% compared with SAGA. It also requires less training time and memory than UDA-GCN on both tasks. FAVA is the fastest method on CN→US, whereas GraphATA is faster on US→CN but requires 5.190 times as much GPU memory.

On ACM1→ACM2 and ACM2→ACM1, FAVA requires longer training times than the three baselines but achieves the lowest peak GPU memory. Compared with SAGA, its training time increases by 9.5% and 13.1%, respectively, while peak GPU memory decreases by 30.9% and 32.2%. UDA-GCN is the fastest method on both ACM tasks, but its memory usage is approximately 1.9 times that of FAVA.

Overall, FAVA achieves consistent memory savings across all four tasks. Its training-time advantage is evident over SAGA and UDA-GCN on the two MAG tasks, while the ACM results show a trade-off between training time and memory usage.

D DESCRIPTION OF THE FAVA ALGORITHM

Algorithm 1 summarizes the training and inference procedures of FAVA. The graph-filtering operations are performed before training because they contain no learnable parameters. During training, FAVA encodes the LP, HP, and FP filtered signals for each relation view and characterizes the frequency composition with the normalized channel-variance vector p_v^u. Corresponding source-target views are assigned a fixed alignment weight of 1, while non-corresponding view pairs receive adaptive weights based on frequency-composition similarity. The stop-gradient operation treats the adaptive weights as constants during backpropagation and introduces no additional learnable parameters. FAVA then updates the model parameters using the prediction and alignment losses. For inference, only the target graph and the trained model are required.

Algorithm 1 The proposed FAVA algorithm

Input: Source graph $G^S = (\mathcal{N}^S, X^S, \{A_v^S\}_{v\in\mathcal{M}}, Y^S)$; target graph $G^T = (\mathcal{N}^T, X^T, \{A_v^T\}_{v\in\mathcal{M}})$; number of training epochs E.
Output: Target-domain predictions $\widehat{Y}^T$.

for $u \in \{S, T\}$ **do**
  for $v \in \mathcal{M}$ **do**
    Construct the normalized adjacency matrix $\widehat{A}_v^u$
    Obtain the LP, HP, and FP filtered signals $X_{v,\mathrm{LP}}^u$, $X_{v,\mathrm{HP}}^u$, and $X_{v,\mathrm{FP}}^u$
  end for
end for
for $e = 1, \ldots, E$ **do**
  for $u \in \{S, T\}$ **do**
    for $v \in \mathcal{M}$ **do**
      Encode the LP, HP, and FP channel representations $Z_{v,\mathrm{LP}}^u$, $Z_{v,\mathrm{HP}}^u$, and $Z_{v,\mathrm{FP}}^u$
      Construct the normalized channel-variance vector p_v^u
    end for
  end for
  for $v \in \mathcal{M}$ **do**
    for $w \in \mathcal{M}$ **do**
      Compute the frequency-composition similarity o_{vw}
      Assign the alignment weight ω_{vw}
    end for
  end for
  Compute the cross-view frequency alignment loss $\mathcal{L}_a$
  for $u \in \{S, T\}$ **do**
    Concatenate the channel and view representations into Z^u
    Obtain the class probabilities Q^u
  end for
  Compute the source classification loss $\mathcal{L}_s$ and target entropy loss $\mathcal{L}_t$
  Compute the overall objective $\mathcal{L} = \mathcal{L}_s + \lambda_t \mathcal{L}_t + \lambda_a \mathcal{L}_a$
  Update all learnable parameters according to $\mathcal{L}$
end for
Predict $\widehat{y}_i^T = \arg\max_c Q_{i,c}^T$ for each $i \in \mathcal{N}^T$
Return: $\widehat{Y}^T = \{\widehat{y}_i^T\}_{i=1}^{n^T}$

For each relation view, FAVA separately encodes the LP, HP, and FP filtered signals and summarizes their relative representation variation using the normalized channel-variance vector $\mathbf{p}_v^u$. Corresponding source-target views receive a fixed alignment weight of 1, whereas each non-corresponding view pair receives an adaptive weight determined by its frequency-composition similarity. The stop-gradient operation prevents the adaptive weights from being directly optimized and introduces no additional learnable parameters. At inference time, only the target graph and the trained encoders and classifier are required.

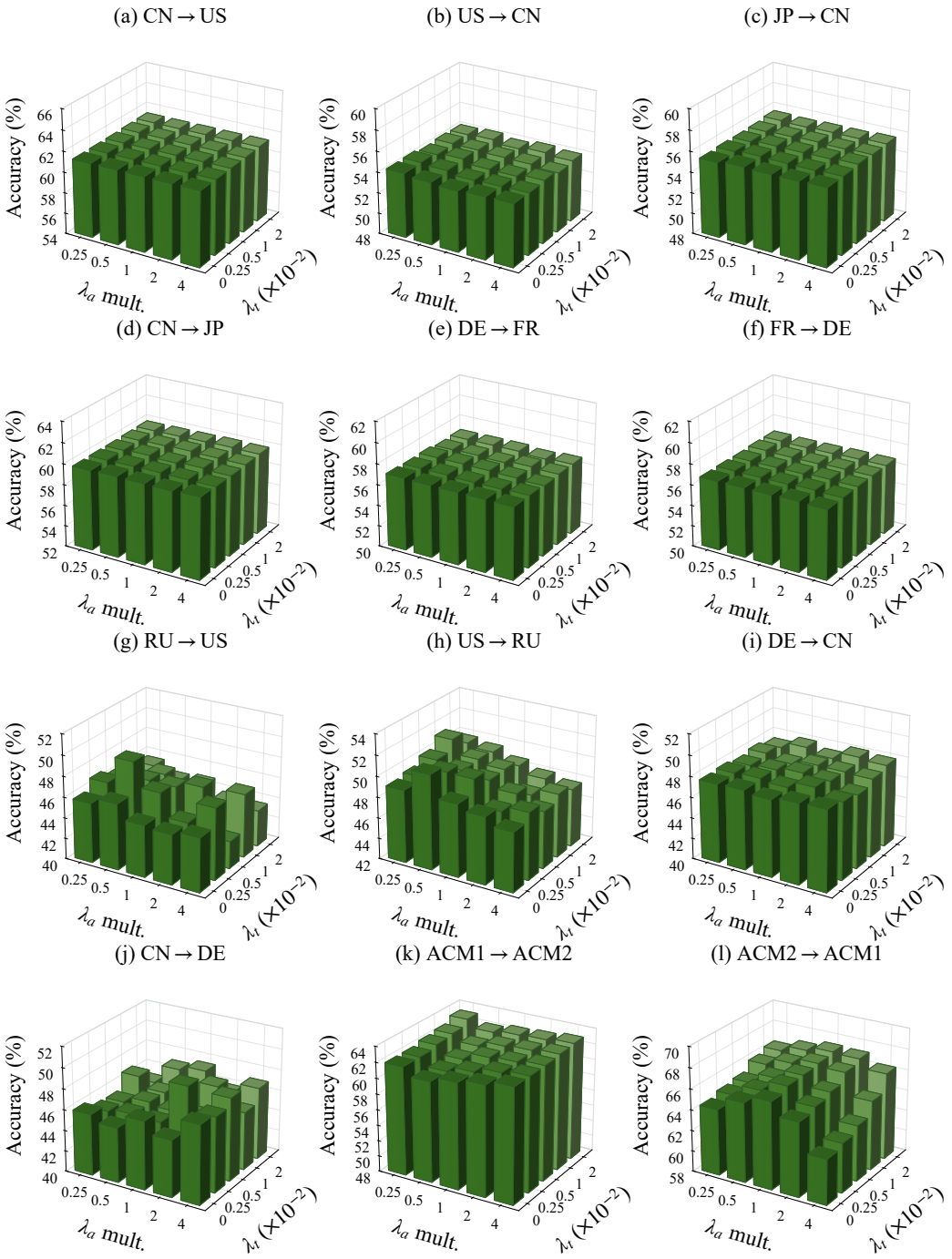


Figure 3: Parameter sensitivity of FAVA. Mean Accuracy (%) under different composition-alignment multipliers m_{a} and target conditional entropy weights λ_{t}. The multiplier $lambda_{\mathrm{a}}$ scales the task-specific λ_{t} reported in Table 2.

E Parameter Analysis

Table 2 reports the hyperparameter settings used for the individual transfer tasks. The parameter λ_t controls the contribution of target-domain conditional entropy to prediction, while λ_a determines the strength of cross-view frequency alignment between source and target views.

Table 2: Hyperparameter settings for individual transfer tasks.

Dataset	Transfer Task	λ_{a}	λ_{t}
ACM	ACM1→ACM2	80	0.0025
	ACM2→ACM1	20	0.0025
MAG	CN→US	80	0.01
	US→CN	160	0.01
	JP→CN	80	0.01
	CN→JP	10	0.01
	DE→FR	2	0.01
	FR→DE	40	0.01
	RU→US	40	0.01
	US→RU	5	0.01
	DE→CN	10	0.01
	CN→DE	20	0.01

We examine the sensitivity of FAVA to λ_a and λ_t. For each transfer task, the task-specific λ_a in Table 2 is scaled by $m_a \in \{0.25, 0.5, 1, 2, 4\}$, while λ_t is varied over $\{0, 0.0025, 0.005, 0.01, 0.02\}$. All other settings remain unchanged. Each parameter configuration is evaluated with three random seeds, and the mean Accuracy is reported.

Figure 3 shows that FAVA maintains stable performance across a broad range of parameter settings on most transfer tasks. Some tasks, including the ACM transfers, exhibit greater sensitivity, but strong performance is still obtained under multiple neighboring configurations rather than a single parameter choice. Moderate cross-view frequency alignment generally yields better results on these tasks, while the preferred target entropy weight varies across transfer directions. Overall, the results indicate that FAVA is not highly dependent on careful hyperparameter tuning.

F Ablation Results

The ablation settings are described in the main text. Here, we report the complete results for all ten MAG transfer tasks.

As shown in Figure 4, Full FAVA achieves the highest Accuracy on all ten transfer tasks. Removing frequency alignment consistently reduces performance, with decreases ranging from 1.2 percentage points on CN→DE to 7.0 percentage points on CN→US. Restricting alignment to corresponding source-target views also leads to a consistent decrease of 0.4-2.9 percentage points, showing the benefit of including non-corresponding view pairs. Among the single-channel variants, LP Only performs best on all ten tasks, but still remains below Full FAVA. Overall, neither a single channel nor alignment restricted to corresponding views matches the performance of the full model.

G Node Embedding Visualization

We further visualize the target-domain node embeddings learned by FAVA on all ten MAG transfer tasks using t-SNE. As shown in Figure 5, nodes from the same class generally form compact clusters, while different classes are separated to varying degrees across transfer directions. Clear class structures can be observed on most tasks, although some overlap remains in more difficult transfer settings. These visualizations provide an additional view of the representations learned by FAVA across different source-target domains.

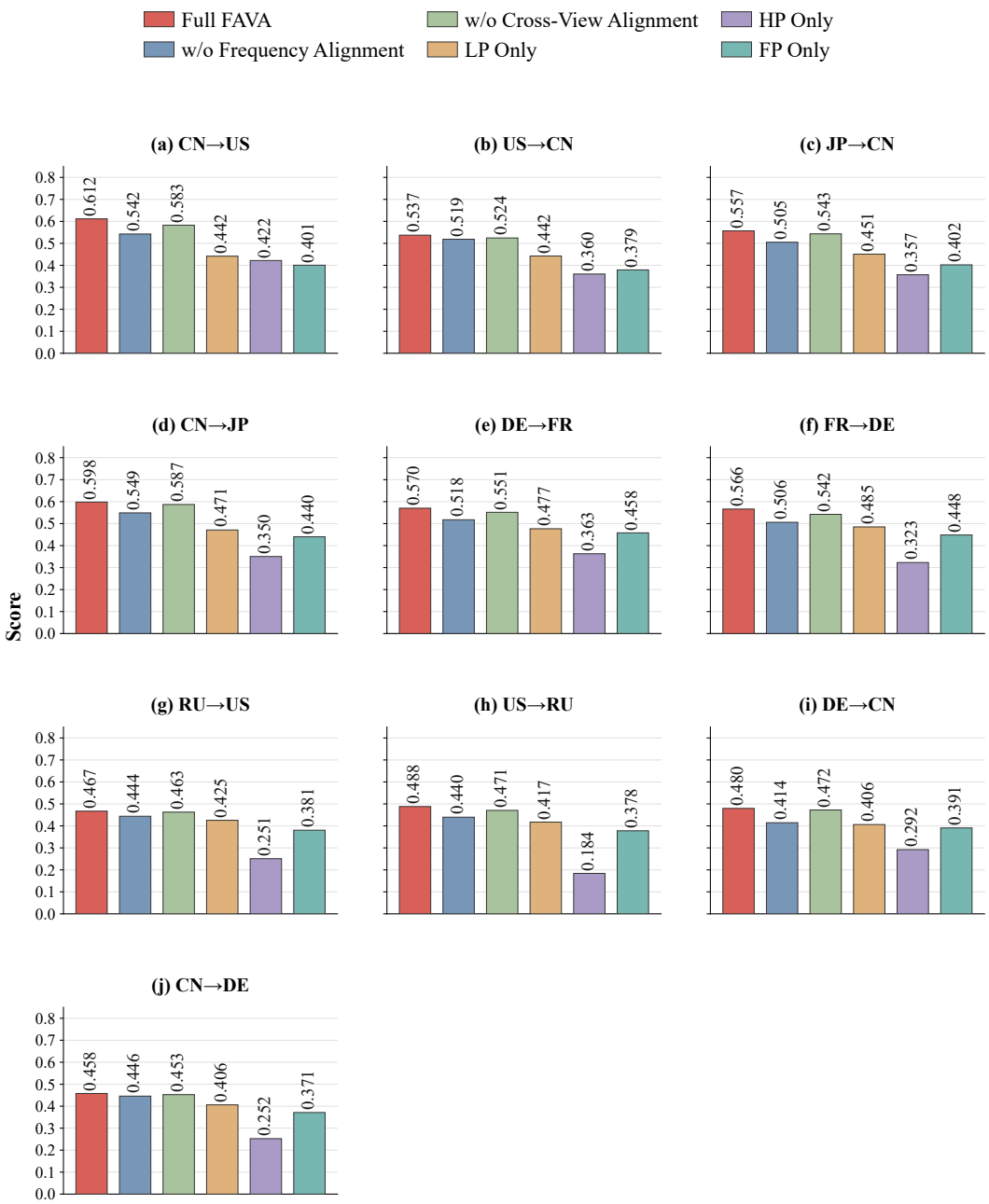


Figure 4: Complete ablation results on MAG. Each panel corresponds to one transfer direction.

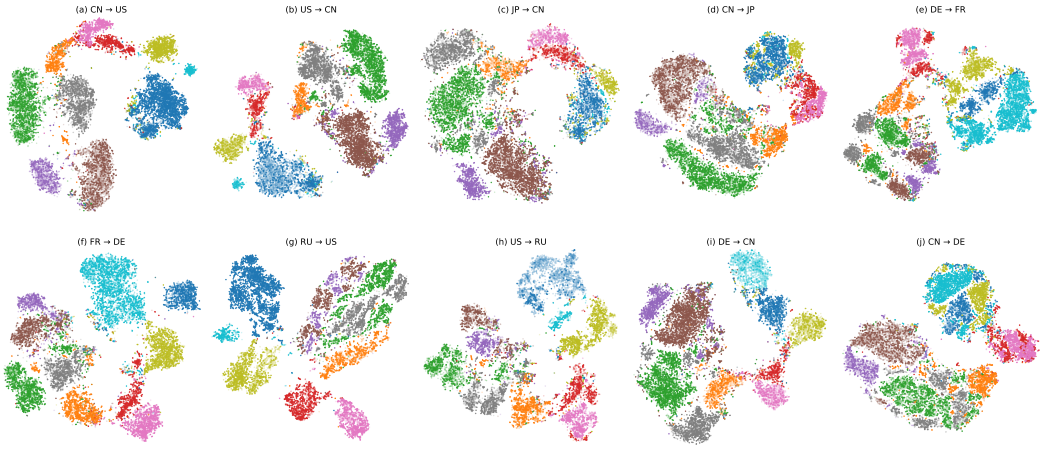


Figure 5: Node embedding visualization on MAG. t-SNE visualizations of the target-domain node embeddings learned by FAVA on all ten MAG transfer tasks. Colors denote different node classes.