BARLOW TWINS-GUIDED ADAPTIVE ALIGNMENT FOR MULTI-VIEW GRAPH DOMAIN ADAPTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-view graph domain adaptation (MGDA) transfers knowledge from a labeled source graph to an unlabeled target graph when both are observed through multiple views. Because domain shift differs across views, aligning the two graphs with a single view-agnostic objective is insufficient, yet existing methods weight all view pairs uniformly and overlook how each source view relates to its target counterpart. We characterize this relationship for every source-target view pair with two quantities: consistency and redundancy. Our empirical results reveal that domain discrepancy decreases with consistency and increases with redundancy. This suggests these two quantities can serve as both training signals and indicators of how strongly to align each pair. We therefore propose Barlow Twins-Guided Adaptive Alignment (BTGA), which applies the invariance and redundancy-reduction terms of Barlow Twins to each source-target view pair and uses the resulting consistency and redundancy scores to weight the pair's alignment. Experimental results on a variety of benchmark datasets verify the effectiveness of our method.

1 INTRODUCTION

Real-world data often involve multiple types of relations among the same set of entities (Qu et al., 2017; Xia et al., 2025). Multi-view graphs provide a natural representation of such data, with each view describing one type of interaction among a common set of nodes (Yu et al., 2024; Yang et al., 2023). They are prevalent in real-world applications, including citation networks, social networks, and knowledge graphs (Wu et al., 2023b; Pan & Kang, 2023; Yu et al., 2025; Zhuo et al., 2025). However, annotating graph-structured data is often difficult and costly, leading to pervasive label scarcity (Xu et al., 2022; Huang et al., 2025a; Chen et al., 2019; Shi et al., 2025). To mitigate this problem, multi-view graph domain adaptation (MGDA) transfers knowledge from labeled multi-view source graphs to unlabeled multi-view target graphs (Fang et al., 2026).

In multi-view graphs, each view encodes a distinct type of relational information, leading to varying distribution shifts across views between source and target domains. Such view-level discrepancies

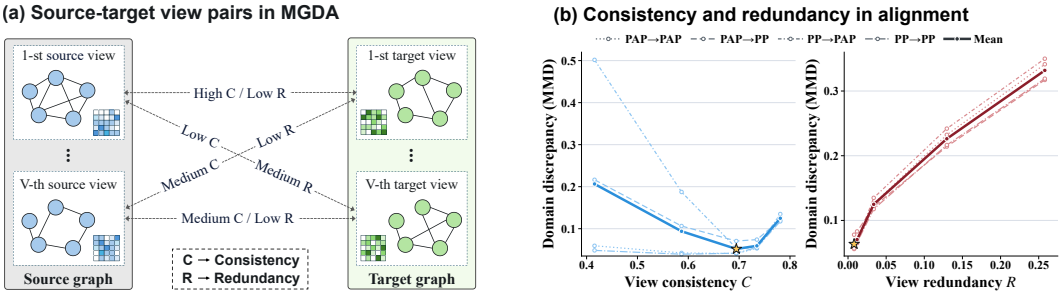


Figure 1: Multi-view graph domain adaptation and empirical analysis of consistency and redundancy. (a) MGDA involves cross-domain alignment across multiple relation views, yielding multiple source-target view pairs, each associated with view consistency and representation redundancy. (b) Across view pairs, domain discrepancy measured by Maximum Mean Discrepancy (MMD) generally decreases as view consistency increases, whereas it generally increases with view redundancy. These results are illustrated on the US→CN transfer task.

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.

pose a challenge for domain alignment in MGDA, where a single view-agnostic objective is insufficient for aligning source and target representations across different views. However, existing methods typically treat different views equally during alignment (Zhu et al., 2025; Xu et al., 2025) and overlook how each source view relates to the target views (Fang et al., 2026).

In this work, we characterize the relationship for every source-target view pair with two quantities: *consistency* and *redundancy*, as illustrated in Figure 1(a). Moreover, our empirical results reveal that view-level domain discrepancy decreases with consistency and increases with redundancy, as shown in Figure 1(b). These observations suggest that the two quantities can serve as effective indicators of discrepancy among source-target view pairs. Specifically, high consistency implies shared information across views (Lyu et al., 2022; Pan & Kang, 2021), which facilitates learning domain-invariant representations. Reduced redundancy further helps preserve feature diversity for effective alignment (Mo et al., 2023a; Shen et al., 2024b).

Based on this insight, we propose **B**arlow **T**wins-**G**uided **A**daptive Alignment (**BTGA**), a framework that achieves adaptive alignment across source-target view pairs. BTGA employs a feature-aware graph filter to integrate view-specific topology and node attributes, obtaining informative view-specific representations. Across domains, BTGA applies the invariance and redundancy-reduction terms of Barlow Twins to each source-target view pair and converts the resulting consistency and redundancy scores into pair-specific alignment weights.

Our main contributions are summarized as follows:

- We reveal that consistency and redundancy provide effective guidance for reducing view-level domain discrepancy. Empirical results show that domain discrepancy decreases with consistency and increases with redundancy.
- We propose Barlow Twins-Guided Adaptive Alignment (BTGA), which enhances cross-view consistency and reduces representation redundancy while deriving pair-specific alignment weights for adaptive cross-domain alignment.
- Experiments on twelve transfer tasks across ACM and MAG benchmarks show that BTGA consistently outperforms representative GDA and MGDA baselines.

2 Related Work

2.1 Multi-view Graph Learning

Multi-relational graphs consist of nodes connected by multiple types of relations, with different relations describing different interactions among nodes (Qu et al., 2017). Multi-view graph learning aims to learn useful representations from multiple relational structures and has received increasing attention (Shen et al., 2024b; Huang et al., 2025b; Zhu et al., 2026; Hwang et al., 2021). Early methods such as MvAGC and MCGC use graph filtering to integrate attribute and structural information from multiple views (Lin et al., 2021; Pan & Kang, 2021). With the advent of deep learning, numerous methods learn node representations via GNNs and self-supervised objectives (Mo et al., 2023b; Peng et al., 2023). BTGF designs a graph filter motivated by theoretical analysis of Barlow Twins to better account for correlations among multiple graphs (Qian et al., 2024).

2.2 Graph Domain Adaptation

Graph domain adaptation (GDA) transfers knowledge from labeled source graphs to unlabeled or sparsely labeled target graphs under distribution shifts (Zhuang et al., 2020; Li et al., 2025). Early methods combine graph encoders with adversarial learning or distribution matching to learn transferable node representations (Zhang et al., 2019; Shen et al., 2020; Wu et al., 2020; Zhang et al., 2021; Dai et al., 2022; Pang et al., 2023; Qiao et al., 2023). Subsequent work addresses structural shifts through graph reweighting, hierarchy modeling, spectral regularization, propagation redesign, pairwise edge and label alignment, spectral disentanglement, and progressive adjustment (Liu et al., 2023; Shi et al., 2023; You et al., 2022; Wu et al., 2023a; Liu et al., 2024a;c; Yang et al., 2025; Wen et al., 2026). Other studies consider attribute and homophily shifts, distribution matching, structural-semantic evolution, interpolation, and gradual adaptation (Fang et al., 2025a;b; Chen et al., 2026a; Lei et al., 2026; Chen et al., 2026b; Qiao et al., 2024). Open-set and unsupervised settings motivate variational exploration, mixup, dual reprogramming, and causal intervention (Yin et al., 2024; 2025;

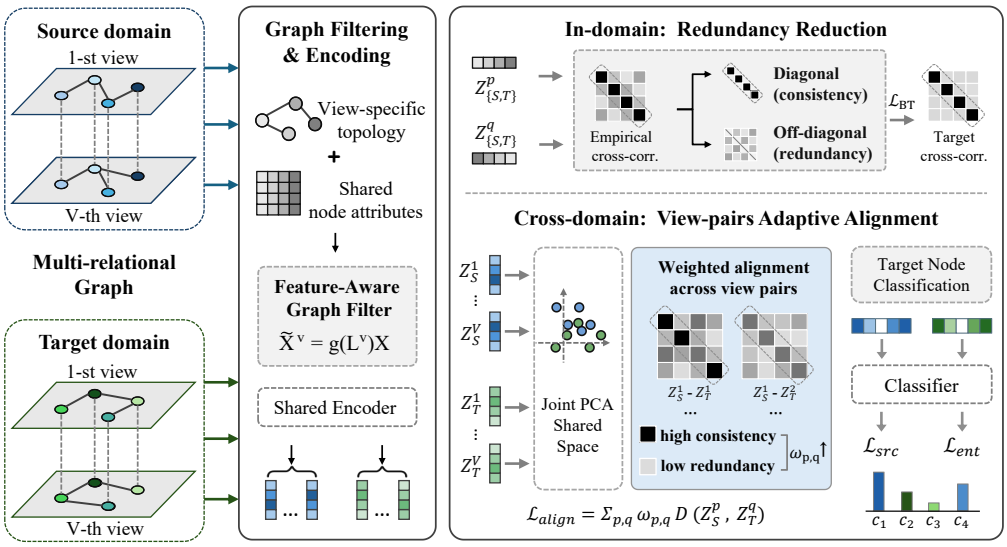


Figure 2: Overview of BTGA. BTGA learns view-specific representations, reduces intra-domain redundancy with Barlow Twins, and incorporates cross-domain consistency and redundancy into training to adaptively weight source-target view-pair alignment.

Zhang & He, 2026; Luo et al., 2025). For class-imbalanced, source-free, and multi-source settings, methods use prototypes, contrastive learning, pseudo-supervision, ranking, bidirectional adaptation, Gaussian mixtures, or hierarchical alignment (Ma et al., 2025; 2026; Zhang et al., 2026; Luo et al., 2026; 2024; Zhang et al., 2024b; Wang et al., 2025; Shen et al., 2026; Lin et al., 2025). In multi-view GDA, SAGA selects a view-order pair for cross-domain alignment (Fang et al., 2026).

3 Methodology

As illustrated in Figure 2, BTGA consists of three stages: (i) learning view-specific node representations through graph filtering and a shared encoder, (ii) reducing representation redundancy across relation views within each domain, and (iii) characterizing source-target view pairs by consistency and redundancy to derive adaptive alignment weights. The resulting representations are then used for target-node classification.

3.1 Problem Setting

We consider an undirected multi-relational graph with V relation views. A domain $D \in \{S, T\}$ is represented as $\mathcal{G}^D = \{\mathcal{V}^D, \{A^{D,v}\}_{v=1}^V, X^D, Y^D\}$, where $\mathcal{V}^D$ is the node set, $A^{D,v} \in \mathbb{R}^{N_D \times N_D}$ is the adjacency matrix of view v, $X^D \in \mathbb{R}^{N_D \times F}$ is the node-attribute matrix, and $Y^D \in \{0,1\}^{N_D \times C}$ is the node-label matrix when available. We are given a labeled source graph $\mathcal{G}^S = \{\mathcal{V}^S, \{A^{S,v}\}_{v=1}^V, X^S, Y^S\}$ and an unlabeled target graph $\mathcal{G}^T = \{\mathcal{V}^T, \{A^{T,v}\}_{v=1}^V, X^T\}$, where $N_D = |\mathcal{V}^D|$. The two domains share the same label space, but their node sets and node counts may differ. The goal is to learn a classifier from the labeled source graph and unlabeled target graph for target-node classification.

3.2 Multi-view Graph Encoding

Node attributes are shared across relation views, while graph structures vary across views. To incorporate topology of each relation view into node attributes, we apply graph filtering to each view (Qian et al., 2024). For domain $D \in \{S, T\}$, we define $\widetilde{A}^{D,v} = A^{D,v} + I$ and $\widetilde{D}_{i,i}^{D,v} = \sum_j \widetilde{A}_{i,j}^{D,v}$. The normalized adjacency matrix is $\widehat{A}^{D,v} = \left(\widetilde{D}^{D,v}\right)^{-1/2} \widetilde{A}^{D,v} \left(\widetilde{D}^{D,v}\right)^{-1/2}$, and the graph Laplacian is $L^{D,v} = I - \widehat{A}^{D,v}$, where I is the identity matrix. Considering the structural variation across

relation views, we learn a graph filter $K^{D,v}$ for each view by solving

$$\min_{K^{D,v}} \left\|X^D - K^{D,v}X^D\right\|_F^2 + \gamma \left\|K^{D,v} - \phi\big(L^{D,v}\big)\right\|_F^2 \tag{1}$$

where $\gamma > 0$ is a trade-off parameter and $\phi\big(L^{D,v}\big) = \left(I - \frac{L^{D,v}}{2}\right)^k$ is a k-th-order low-pass graph filter. The first term preserves the node attributes, while the second term constrains the filter using the topology of the corresponding relation view. The objective in Eq. (1) can be solved by setting its first-order derivative with respect to $K^{D,v}$ to zero, yielding the optimal graph filter

$$K^{D,v} = \left[\gamma\phi\big(L^{D,v}\big) + X^D(X^D)^\top\right]\left(X^D(X^D)^\top + \gamma I\right)^{-1}. \tag{2}$$

Details of the efficient graph filter computation are provided in Appendix C.2. With the learned filter $K^{D,v}$, the filtered node attributes are obtained as $\widetilde{X}^{D,v} = K^{D,v}X^D$, which integrate the topology of relation view v with the shared node attributes. They are then fed into an encoder shared across relation views and domains:

$$Z^{D,v} = f_\Theta\left(\widetilde{X}^{D,v}\right) \tag{3}$$

where $Z^{D,v} \in \mathbb{R}^{N_D \times d}$ denotes the node representation of the v-th relation view in domain D, and d is the embedding dimension. The shared encoder maps all relation views and both domains into a common latent space for subsequent cross-view redundancy reduction and cross-domain alignment.

3.3 Intra-domain Cross-view Redundancy Reduction

To reduce representation redundancy across relation views within each domain, we regularize the learned view representations through feature decorrelation. Specifically, Barlow Twins (BT) reduces redundancy by driving the cross-correlation matrix toward the identity matrix. For two view-specific representations $Z^{D,p}$ and $Z^{D,q}$ from the same domain, we define their cross-correlation matrix as $M^{D,pq} \in \mathbb{R}^{d\times d}$, whose (i,j)-th element is

$$M_{i,j}^{D,pq} = \frac{\sum_b z_{b,i}^{D,p} z_{b,j}^{D,q}}{\sqrt{\sum_b \left(z_{b,i}^{D,p}\right)^2}\sqrt{\sum_b \left(z_{b,j}^{D,q}\right)^2}} \tag{4}$$

where b indexes nodes and i, j index representation dimensions. The normalized BT loss is

$$\mathcal{L}_{\mathrm{BT}}\big(Z^{D,p}, Z^{D,q}\big) = \frac{1}{d}\sum_i \left(1 - M_{i,i}^{D,pq}\right)^2 + \frac{1}{d(d-1)}\sum_i\sum_{j\neq i}\left(M_{i,j}^{D,pq}\right)^2 \tag{5}$$

where d is the representation dimension. The BT objective reduces feature redundancy by suppressing off-diagonal correlations between different representation dimensions. We apply BT to every unordered pair of relation views within each domain. The resulting intra-domain objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{intra}} = &\frac{2}{V(V-1)}\sum_{1\le p<q\le V}\mathcal{L}_{\mathrm{BT}}\big(Z^{S,p}, Z^{S,q}\big)\\ &+\frac{2}{V(V-1)}\sum_{1\le p<q\le V}\mathcal{L}_{\mathrm{BT}}\big(Z^{T,p}, Z^{T,q}\big)\end{aligned} \tag{6}$$

where, p and q denote two distinct relation views. In each domain, V views yield $V(V-1)/2$ unordered view pairs. By averaging the pairwise BT losses within each domain, $\mathcal{L}_{\mathrm{intra}}$ reduces redundant feature correlations across relation views. Pairing every source view with every target view yields V^2 cross-domain view pairs, as considered next.

3.4 Adaptive Cross-domain Alignment

Cross-domain View-pair Correlation. Since source and target graphs contain unpaired nodes, cross-domain correlations cannot be computed directly from their view-specific node representations. Therefore, we summarize multi-view representations of both domains in a shared r-dimensional space, yielding a fixed-size statistical representation $H^{D,v} \in \mathbb{R}^{2r\times d}$ for each view v in domain $D \in \{S, T\}$. Its $2r$ rows correspond to positive and negative summaries along the

r shared principal directions. We standardize each column before computing cross-domain correlations. Specifically, we use column-standardized entries $\widetilde{H}^{D,v}_{\ell,i} = (H^{D,v}_{\ell,i} - \mu^{D,v}_i)/(\sigma^{D,v}_i + \epsilon)$, where $\mu^{D,v}_i$ and $\sigma^{D,v}_i$ are the mean and standard deviation of column i of $H^{D,v}$, respectively. The cross-correlation matrix between source view p and target view q is then

$$\mathbf{M}^{p,q} = \frac{1}{2r}\left(\widetilde{H}^{S,p}\right)^{\top} \widetilde{H}^{T,q} \tag{7}$$

where $\mathbf{M}^{p,q} \in \mathbb{R}^{d\times d}$ characterizes their cross-domain correlation structure.

Consistency and Redundancy Guided View-Pair Weighting. Following Barlow Twins, diagonal correlations reflect the agreement between corresponding dimensions, whereas off-diagonal correlations characterize dependencies across different dimensions. Based on $\mathbf{M}^{p,q}$, we define the consistency of source-target view pair (p, q) as

$$C_{p,q} = 1 - \frac{1}{d}\sum_{i=1}^{d}\left(1 - \mathbf{M}^{p,q}_{i,i}\right)^2 \tag{8}$$

and its representation redundancy as

$$R_{p,q} = \frac{1}{d(d-1)}\sum_{i=1}^{d}\sum_{j\neq i}\left(\mathbf{M}^{p,q}_{i,j}\right)^2 \tag{9}$$

As the diagonal correlations approach one, $C_{p,q}$ increases, indicating stronger agreement between corresponding dimensions. In contrast, suppressing the off-diagonal correlations toward zero decreases $R_{p,q}$, reducing redundancy across different dimensions. We combine these two quantities into a view-pair score:

$$s_{p,q} = (1 - C_{p,q}) + \lambda_{\text{inter}} R_{p,q} \tag{10}$$

where $\lambda_{\text{inter}} > 0$ controls the contribution of redundancy relative to consistency. We parameterize $\lambda_{\text{inter}} = \text{softplus}(\rho)$ to ensure positivity, where ρ is jointly optimized with the model parameters. A source-target view pair with higher consistency and lower redundancy therefore receives a smaller score. We convert the scores into normalized alignment weights as

$$\omega_{p,q} = \frac{\exp(-s_{p,q}/\tau)}{\sum_{a=1}^{V}\sum_{b=1}^{V}\exp(-s_{a,b}/\tau)} \tag{11}$$

where $\tau > 0$ controls the sharpness of the weight distribution. Accordingly, view pairs with smaller scores receive larger weights and contribute more strongly to cross-domain alignment. The correlation statistics are computed from detached full-domain representations and refreshed once per epoch, while λ_{inter} remains differentiable through $\omega_{p,q}$.

Weighted Cross-domain Alignment. We measure distribution discrepancy using multi-kernel maximum mean discrepancy (MK-MMD), which combines multiple kernels to provide a flexible measure of cross-domain distribution discrepancy (Gretton et al., 2012; Zhang et al., 2024a). Specifically, $\text{MK-MMD}^2(P, Q) = \sum_{\ell=1}^{L}\eta_\ell \,\text{MMD}^2_{k_\ell}(P, Q)$, where k_ℓ is the ℓ-th kernel, $\eta_\ell \geq 0$, $\sum_{\ell=1}^{L}\eta_\ell = 1$, and L is the number of kernels. We then align source and target view distributions using the correlation-weighted MK-MMD objective:

$$\mathcal{L}_{\text{align}} = \sum_{p=1}^{V}\sum_{q=1}^{V}\omega_{p,q}\ \text{MK-MMD}^2\left(Z^{S,p}, Z^{T,q}\right). \tag{12}$$

3.5 Target Node Classification

For node classification, the view-specific representations are aggregated in each domain into Z^D, where $D \in \{S, T\}$. The resulting representation Z^D is fed into a classifier $f_\Phi(\cdot)$ shared across domains to obtain the class probabilities $\widehat{Y}^D = \text{softmax}\left(f_\Phi(Z^D)\right)$. The source classification loss $\mathcal{L}_S$ minimizes the cross-entropy over the labeled nodes in the source domain:

$$\mathcal{L}_S = -\frac{1}{N_S}\sum_{i=1}^{N_S}\sum_{c=1}^{C} Y^S_{ic}\log\widehat{Y}^S_{ic}. \tag{13}$$

where Y_{ic}^S denotes the ground-truth label of the i-th source node for class c, and $\widehat{Y}_{ic}^S$ is the corresponding predicted probability. To utilize the unlabeled target nodes, the entropy of the target predictions is further minimized by minimizing:

$$\mathcal{L}_T = -\frac{1}{N_T}\sum_{i=1}^{N_T}\sum_{c=1}^{C}\widehat{Y}_{ic}^T \log \widehat{Y}_{ic}^T. \tag{14}$$

where $\widehat{Y}_{ic}^T$ denotes the predicted probability that the i-th target node belongs to class c. Finally, the overall training objective combines the source classification, intra-domain redundancy reduction, cross-domain alignment, and target entropy losses:

$$\mathcal{L} = \mathcal{L}_S + \mathcal{L}_{\text{intra}} + \alpha\mathcal{L}_{\text{align}} + \beta\mathcal{L}_T \tag{15}$$

where α and β are trade-off hyperparameters controlling the contributions of cross-domain alignment and target entropy minimization, respectively. The encoder f_Θ, classifier f_Φ, and the learnable parameter ρ are jointly optimized via backpropagation.

4 EXPERIMENT

4.1 DATASETS

We use real-world citation network datasets to evaluate our method for multi-view graph domain adaptation in the node classification task. Specifically, we conduct experiments on ACM1 (Shen et al., 2024a), ACM2 (Shen et al., 2024a), and MAG (Shen et al., 2024a) to evaluate the effectiveness of knowledge transfer across views and domains while preserving node-class information.

In these graphs, nodes represent papers and edges represent relations between papers. Each graph contains multiple views capturing different types of relations. ACM1 and ACM2 contain two views defined by the PAP (paper-author-paper) and PSP (paper-subject-paper) meta-paths. MAG contains two views defined by the PAP (paper-author-paper) and PP (paper-paper) meta-paths.

The ACM1 and ACM2 graphs are labeled into three classes: database, wireless communication, and data mining. As the subset of OGBN-MAG (Wang et al., 2020), MAG is partitioned into six subgraphs (CN, US, DE, FR, JP, and RU), each corresponding to a country. From these country subgraphs, we extract the ten most frequent classes. Any two of these country subgraphs can be used as the source and target datasets in the multi-view graph node classification experiment. Detailed statistics of the experimental datasets are provided in Appendix A.

4.2 BASELINES

In this work, we compare a representative set of baselines for multi-view graph domain adaptation (GDA) in node classification. **GCN** (Kipf & Welling, 2016) provides a standard non-adaptive baseline by performing graph convolution without any domain alignment objective, while **GAT** (Veličković et al., 2018) offers an attention-based alternative that replaces uniform aggregation with learned neighbor weighting. **AdaGCN** (Dai et al., 2022) performs adversarial graph domain adaptation by adapting graph convolutional representations across views. **UDAGCN** (Wu et al., 2020) captures local and global graph information and aligns representations across views through adversarial training. **GRADE-N** (Wu et al., 2023a) introduces a graph subtree discrepancy to quantify distribution shifts between source and target graphs across relation views. **ACDNE** (Shen et al., 2020) learns transferable representations across networks through adversarial domain adaptation applied to each relation view, while **ASN** (Zhang et al., 2021) separates domain-private and domain-shared information to extract transferable representations.

Table 1: Cross-network node classification results on ACM datasets.

Methods	ACM1→ACM2		ACM2→ACM1	
	ACC	F1	ACC	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
UDAGCN	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	<u>0.523</u>	<u>0.351</u>	<u>0.454</u>	<u>0.444</u>
BTGA	**0.666**	**0.664**	**0.687**	**0.684**

Table 2: Cross-network node classification results on MAG datasets.

Methods	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
UDAGCN	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	0.404	0.255	0.364	0.242	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	0.349	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	0.237	0.331	0.295	0.243	0.229	0.299	0.235	0.367	0.336	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	0.341	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	0.230	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
BTGA	**ACC**	**0.592**	**0.558**	**0.546**	**0.595**	**0.560**	**0.564**	**0.430**	**0.477**	**0.461**	**0.412**
	Macro-F1	**0.410**	**0.455**	**0.414**	**0.463**	**0.494**	**0.493**	**0.418**	**0.436**	**0.390**	**0.394**

SpecReg (You et al., 2022) improves cross-domain generalization across relation views through spectral regularization grounded in graph-filter theory. **PA** (Liu et al., 2024b) mitigates structural and label shifts across relation views by recalibrating edge influences and adjusting the classification objective. **GraphATA** (Zhang & He, 2025) performs node-wise, personalized aggregation of source GNNs for multi-source-free unsupervised adaptation. **HGDA** (Fang et al., 2025b) mitigates cross-domain discrepancies by explicitly aligning homophilic, heterophilic, and attribute signals through specialized graph filters. **SAGA** (Fang et al., 2026) jointly considers relation views and neighborhood orders, using a Structural Aggregation Distance to dynamically identify a dominant view-order pair that guides cross-domain alignment.

4.3 Experimental Setup

The experiments are implemented in PyTorch using an Intel(R) Xeon Platinum 8352V CPU and NVIDIA RTX A6000 48G GPUs. A two-layer MLP encoder is constructed, and the model is trained using the Adam optimizer (Kingma & Ba, 2015), with the learning rate ranging from 0.001 to 0.005. To prevent overfitting, the dropout rate is set to 0.05. In addition, the weight decay coefficient is selected from $\{1e{-}4, \ldots, 5e{-}3\}$. The model adopts a two-stage representation learning architecture, with the hidden dimension set to 256 and the embedding dimension set to 128. For different transfer tasks, the batch size is determined according to the scale of each dataset: ACM-based tasks use a batch size of 1000, whereas MAG-based tasks adopt a larger batch size of 10000. For fairness, all cross-domain node classification methods are evaluated under the same dataset partitioning and experimental protocol, except for their method-specific hyperparameters. We repeatedly train and test our model five times, reporting node classification performance on both the source and target domains, including ACC and Macro-F1. Hyperparameter settings and sensitivity results are provided in Appendix B.

4.4 Results and Analyses

Experimental results are summarized in Table 1 and Table 2, where the best performance is highlighted in **bold** and the second-best performance is underlined. It can be seen that our proposed BTGA outperforms all baseline methods across all twelve transfer tasks, achieving state-of-the-art (SOTA) performance in terms of both ACC and Macro-F1. Specifically, compared with the second-best results, BTGA achieves average improvements of 3.7% and 5.2% in ACC and Macro-F1 on the MAG datasets, respectively, and improves ACC by 14.3% on ACM1→ACM2.

The unsatisfactory performance of GCN (Kipf & Welling, 2016) and GAT (Veličković et al., 2018) indicates that merely transferring knowledge learned from source domain to target domain is inadequate under domain shift, highlighting the importance of effective cross-domain alignment. While existing adaptation methods generally achieve better performance than non-adaptive methods, they still fall short of BTGA, suggesting that reducing cross-domain discrepancy alone is insufficient when domain shifts vary across views. In particular, although SAGA (Fang et al., 2026) explicitly exploits multiple relational views to guide cross-domain alignment, BTGA consistently achieves better performance across all transfer tasks. These results suggest that adaptively weighting source-target view pairs during alignment is important for effectively mitigating view-level domain discrepancies in MGDA.

4.5 Ablation Study

To evaluate the contribution of different components in BTGA, we compare the full model with five variants on ACM and MAG datasets. Specifically, *w/o Filter* removes the graph filtering module, *w/o Intra* removes the intra-domain redundancy-reduction objective, and *w/o Alignment* removes cross-domain alignment. *Consistency Only* and *Redundancy Only* use only the corresponding quantity to adaptively weight source-target view pairs during alignment.

As shown in Figure 3, BTGA (Full) performs better than all variants across the evaluated tasks, indicating the effectiveness of each component. Removing cross-domain alignment results in ACC drops of 26 and 5 percentage points on ACM2→ACM1 and RU→US, respectively, demonstrating the importance of cross-domain alignment across different transfer tasks. The performance drops after removing either graph filtering or the intra-domain redundancy-reduction objective further show that effective alignment relies on informative view-specific representations with reduced cross-view redundancy. In addition, both *Consistency Only* and *Redundancy Only* perform worse than the full model, indicating that using the two quantities together provides more effective guidance for adaptively weighting source-target view pairs during alignment. Additional ablation results are provided in Appendix E.

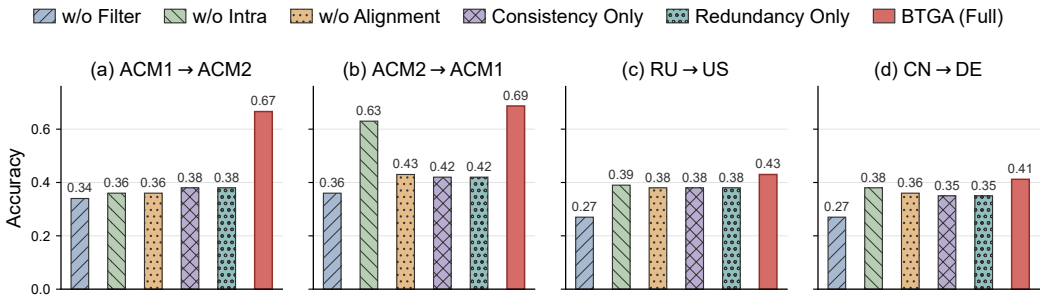
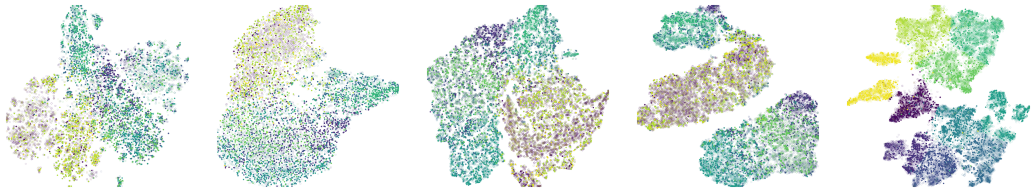


Figure 3: Classification accuracy of BTGA and its ablated variants on ACM and MAG datasets.

4.6 Visualization

In this section, we visualize source and target node embeddings learned by representative graph domain adaptation methods on the RU→US task using t-SNE. As shown in Figure 4, compared with PA, ASN, UDAGCN, and SAGA, our proposed BTGA produces more clearly separated clusters across different classes, indicating more discriminative node representations. Meanwhile, source and target nodes belonging to the same class are largely mixed within the corresponding clusters, suggesting that BTGA effectively reduces cross-domain discrepancy. These observations indicate that BTGA learns representations that are both class-discriminative and well-aligned across domains.



(a) PA (b) ASN (c) UDAGCN (d) SAGA (e) BTGA

Figure 4: t-SNE visualization of source and target node embeddings on the RU→US task. Each color represents a class, while filled and hollow markers denote source and target nodes, respectively.

Additional visualization results are provided in Appendix F.

4.7 Parameter Analysis

In this section, we analyze the sensitivity of the parameters of BTGA on DE→FR and US→CN. As shown in Figure 5(a,d), accuracy reaches its highest value at $k = 2$ on DE→FR and $k = 4$ on US→CN. This is reasonable since increasing k incorporates more high-order neighborhood information. On the other hand, an excessively large k may introduce noise and deteriorate the performance, as observed on DE→FR. Furthermore, each adaptation task may require a different suitable value of k because of its inherent graph properties. From Figure 5(b,c,e,f), BTGA maintains competitive performance over a wide range of α and β, with β being particularly stable on both tasks. Only an excessively large α leads to a noticeable decline on DE→FR. These results demonstrate the stability of BTGA under different parameter settings.

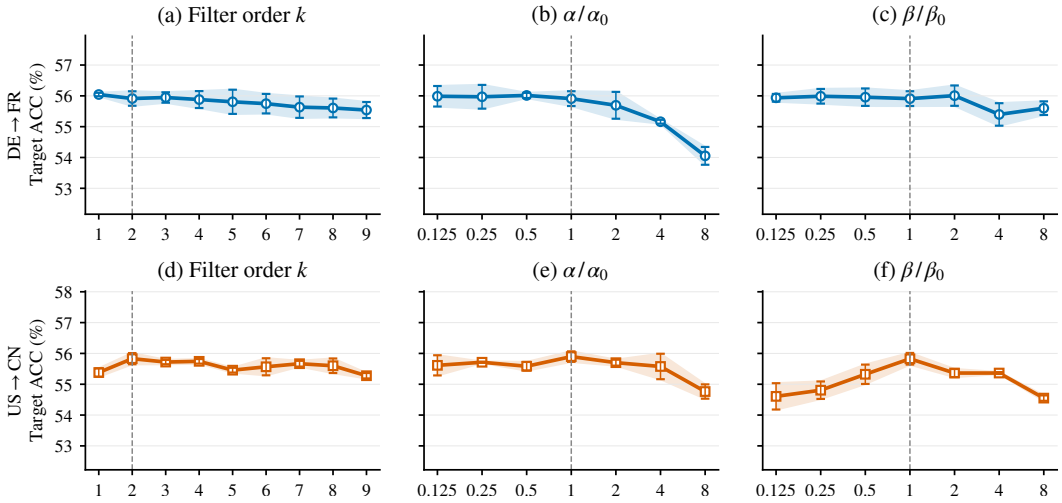


Figure 5: Influence of parameters k, α, and β on DE→FR and US→CN.

5 Conclusion

In this paper, we propose the Barlow Twins-Guided Adaptive Alignment (BTGA) framework to address the multi-view graph domain adaptation problem in cross-domain node classification. Through empirical analysis, we reveal that consistency and redundancy provide effective indicators of view-level domain discrepancy across source-target view pairs. Building on this finding, BTGA incorporates the two signals into training to derive adaptive alignment weights, assigning greater emphasis to view pairs with higher consistency and lower redundancy. Comprehensive experiments on ACM and MAG benchmarks verify the superiority of our approach. We will also delve deeper into multi-view graph domain adaptation theory to develop more powerful and generalizable models.

Use of Generative AI. In this work, we used generative AI tools for language editing, including improving grammar, clarity, and readability. We have not used generative AI tools for research ideation, methodology design, experimental implementation, data analysis, or interpretation of experimental results, and the remaining disclosure tasks are not applicable to this work. We have reviewed all AI-assisted work to ensure that the revisions preserve the intended technical meaning and accurately reflect the authors original research. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

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 *International Conference on Learning Representations*, 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.

Yiming Chen, Shiji Song, Shuang Li, and Cheng Wu. A graph embedding framework for maximum mean discrepancy-based domain adaptation algorithms. *IEEE Transactions on Image Processing*, 29:199–213, 2019.

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, Zhao Kang, Qiuhao Zeng, Nima Hosseini Dashtbayaz, Ruizhi Pu, Boyu Wang, and Charles Ling. On the benefits of attribute-driven graph domain adaptation. *ICLR*, 2025a.

Ruiyi Fang, Bingheng Li, Jingyu Zhao, Ruizhi Pu, Qiuhao Zeng, Gezheng Xu, Charles Ling, and Boyu Wang. Homophily enhanced graph domain adaptation. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 16006–16028, 2025b.

Ruiyi Fang, Jingyu Zhao, Shuo Wang, Ruizhi Pu, Bingheng Li, Jiale Cai, Zhihao Li, Zihao Jing, Jian Zhu, Song Tang, Charles Ling, and Boyu Wang. SAGA: Structural aggregation guided alignment with dynamic view and neighborhood order selection for multiview graph domain adaptation. In *The Fourteenth International Conference on Learning Representations*, 2026.

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.

Jincheng Huang, Yujie Mo, Xiaoshuang Shi, Lei Feng, and Xiaofeng Zhu. Enhancing the influence of labels on unlabeled nodes in graph convolutional networks. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 25613–25633, 2025a.

Yudi Huang, Ci Nie, Hongqing He, Yujie Mo, Yonghua Zhu, Guoqiu Wen, and Xiaofeng Zhu. Multiplex graph representation learning with homophily and consistency. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11835–11842, 2025b.

HyeongJoo Hwang, Geon-Hyeong Kim, Seunghoon Hong, and Kee-Eung Kim. Multi-view representation learning via total correlation objective. *Advances in Neural Information Processing Systems*, 34:12194–12207, 2021.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.

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

Pui Ieng Lei, Ximing Chen, Yijun Sheng, Yanyan Liu, Zhiguo Gong, and Qiang Yang. Gradual domain adaptation for graph learning. *ACM Transactions on Intelligent Systems and Technology*, 17(5):1–45, 2026.

Haoyang Li, Xin Wang, Ziwei Zhang, and Wenwu Zhu. Out-of-distribution generalization on graphs: A survey. *TPAMI*, 2025.

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.

Zhiping Lin, Zhao Kang, Lizong Zhang, and Ling Tian. Multi-view attributed graph clustering. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1872–1880, 2021.

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.

Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural re-weighting improves graph domain adaptation. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 21778–21793. PMLR, 2023.

Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 32552–32575, 2024b.

Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 32552–32575. PMLR, 2024c.

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 *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 4706–4714. International Joint Conferences on Artificial Intelligence Organization, 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.

Qi Lyu, Xiao Fu, Weiran Wang, and Songtao Lu. Understanding latent correlation-based multiview learning and self-supervision: An identifiability perspective. In *The Tenth International Conference on Learning Representations*, 2022.

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.

Yujie Mo, Yuhuan Chen, Yajie Lei, Liang Peng, Xiaoshuang Shi, Changan Yuan, and Xiaofeng Zhu. Multiplex graph representation learning via dual correlation reduction. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12814–12827, 2023a.

Yujie Mo, Yajie Lei, Jialie Shen, Xiaoshuang Shi, Heng Tao Shen, and Xiaofeng Zhu. Disentangled multiplex graph representation learning. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 24983–25005, 2023b.

Erlin Pan and Zhao Kang. Multi-view contrastive graph clustering. *Advances in Neural Information Processing Systems*, 34:2148–2159, 2021.

Erlin Pan and Zhao Kang. High-order multi-view clustering for generic data. *Information Fusion*, 100:101947, 2023.

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.

Liang Peng, Xin Wang, and Xiaofeng Zhu. Unsupervised multiplex graph learning with complementary and consistent information. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 454–462, 2023.

Xiaowei Qian, Bingheng Li, and Zhao Kang. Upper bounding barlow twins: A novel filter for multi-relational clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 14660–14668, 2024.

Ziyue Qiao, Xiao Luo, Meng Xiao, Hao Dong, Yuanchun Zhou, and Hui Xiong. Semi-supervised domain adaptation in graph transfer learning. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pp. 2279–2287. International Joint Conferences on Artificial Intelligence Organization, 2023. doi: 10.24963/ijcai.2023/253.

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.

Meng Qu, Jian Tang, Jingbo Shang, Xiang Ren, Ming Zhang, and Jiawei Han. An attention-based collaboration framework for multi-view network representation learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1767–1776, 2017.

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.

Zhixiang Shen, Haolan He, and Zhao Kang. Balanced multi-relational graph clustering. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 4120–4128, 2024a.

Zhixiang Shen, Shuo Wang, and Zhao Kang. Beyond redundancy: Information-aware unsupervised multiplex graph structure learning. In *Advances in Neural Information Processing Systems*, volume 37, pp. 31629–31658, 2024b.

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.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *ICLR*, 2018.

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.

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*, 2026.

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, 2023a.

Lingfei Wu, Yu Chen, Kai Shen, Xiaojie Guo, Hanning Gao, Shucheng Li, Jian Pei, and Bo Long. Graph neural networks for natural language processing: A survey. *Foundations and Trends in Machine Learning*, 16(2):119–328, 2023b.

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.

Feng Xia, Ciyuan Peng, Jing Ren, Falih Gozi Febrinanto, Renqiang Luo, Vidya Saikrishna, Shuo Yu, and Xiangjie Kong. Graph learning. *Foundations and Trends in Signal Processing*, 19(4): 362–519, 2025.

Zihao Xu, Hao He, Guang-He Lee, Bernie Wang, and Hao Wang. Graph-relational domain adaptation. In *International Conference on Learning Representations*, 2022.

Zitao Xu, Shu Chen, Weike Pan, and Zhong Ming. A multi-view graph contrastive learning framework for cross-domain sequential recommendation. *Transactions on Recommender Systems*, 3(4): 43:1–43:28, 2025. doi: 10.1145/3695884.

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.

Xihong Yang, Yue Liu, Sihang Zhou, Siwei Wang, Wenxuan Tu, Qun Zheng, Xinwang Liu, Liming Fang, and En Zhu. Cluster-guided contrastive graph clustering network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 10834–10842, 2023.

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 adaptation. In *International Conference on Learning Representations*, 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 *The Eleventh International Conference on Learning Representations*, 2022.

Shuo Yu, Huafei Huang, Yanming Shen, Pengfei Wang, Qiang Zhang, Ke Sun, and Honglong Chen. Formulating and representing multiagent systems with hypergraphs. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3):4599–4613, 2024.

Shuo Yu, Yingbo Wang, Zhitao Wan, Yanming Shen, Qiang Zhang, and Feng Xia. Path-aware few-shot knowledge graph completion. *IEEE Transactions on Artificial Intelligence*, 6(8):2133–2147, 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.

Hongliang Zhang, Yuteng Zhang, Rui Wang, Haiyang Pan, and Bin Chen. Non-uniformly weighted multisource domain adaptation network for fault diagnosis under varying working conditions. *Neural Process. Lett.*, 56(2):84, 2024a. doi: 10.1007/S11063-024-11568-2.

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.

Yizhou Zhang, Guojie Song, Lun Du, Shuwen Yang, and Yilun Jin. DANE: Domain adaptive network embedding. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4362–4368, 2019.

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*, 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, 2024b.

Chenyang Zhu, Lanlan Zhang, Weibin Luo, Guangqi Jiang, and Qian Wang. Tensorial multiview low-rank high-order graph learning for context-enhanced domain adaptation. *Neural Networks*, 181:106859, 2025. doi: 10.1016/j.neunet.2024.106859.

Zhengzhong Zhu, Pei Zhou, Dongsheng Wang, Li Cheng, and Jiangping Zhu. Online multi-relational clustering with dominant view mining. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 29232–29240, 2026.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2020.

Jiaming Zhuo, Ziyi Ma, Yintong Lu, Yuwei Liu, Kun Fu, Di Jin, Chuan Wang, Wenning Wu, Zhen Wang, Xiaochun Cao, and Liang Yang. A closer look at graph transformers: Cross-aggregation and beyond. *Advances in Neural Information Processing Systems*, 38:136781–136807, 2025.

A Dataset Details

Table 1 summarizes the statistics of the datasets used in our experiments, including the numbers of nodes, edges, views, and classes.

ACM1 and ACM2 are citation network datasets, where nodes represent papers and edges denote relations between papers. Both datasets contain two views corresponding to the PAP (paper-author-paper) and PSP (paper-subject-paper) meta-paths. The node labels are categorized into three classes: database, wireless communication, and data mining.

MAG is a subset of OGBN-MAG (Wang et al., 2020). We use six country-specific subgraphs, including CN, US, DE, FR, JP, and RU, as different domains. Each subgraph contains paper nodes and two views defined by the PAP (paper-author-paper) and PP (paper-paper) relations. We select the ten most frequent classes from these subgraphs for node classification.

Table 1: Statistics of experimental datasets.

Types	Datasets	#Node	#Relation	#Edge	#Feat	#Label
ACM	ACM1	3,025	PAP	26,256	512	3
			PSP	2,207,736		
	ACM2	4,019	PAP	57,853	512	
			PSP	4,338,213		
MAG	CN	25,467	PAP	157,303	128	10
			PP	43,683		
	US	55,392	PAP	12,332,742	128	
			PP	289,833		
	JP	12,651	PAP	41,165	128	
			PP	20,747		
	FR	12,052	PAP	51,739	128	
			PP	30,915		
	DE	17,919	PAP	77,100	128	
			PP	45,528		
	RU	3,690	PAP	3,315	128	
			PP	3,539		

B Parameter Analysis

Table 2 summarizes the task-specific settings of the four hyperparameters used in BTGA. Here, α and β control the intra-domain objective and cross-domain alignment objective, respectively. The filter order k determines the order of the low-pass graph operator, while γ controls the trade-off between attribute preservation and topology-based filtering. Since the optimal balance between these components can vary with graph structure and domain shift, different settings are adopted for different transfer tasks.

We further investigate the sensitivity of BTGA to α, β, k, and γ over all twelve transfer tasks. For α, β, and γ, each parameter is varied relative to its task-specific default value while the remaining parameters are fixed; the filter order k is varied directly. For clarity, the transfer tasks are divided into three groups across the columns of Figure 1.

As shown in Figure 1, the sensitivity of BTGA is task dependent. For α, most MAG tasks exhibit relatively small variations, while ACM2→ACM1 shows a clear optimum around the default setting and ACM1→ACM2 gradually degrades as α increases. A similar pattern is observed for β: several tasks remain stable over a broad range, whereas excessively large values noticeably reduce performance on ACM2→ACM1 and DE→CN. The filter order k has a more pronounced effect on the ACM tasks, where increasing k substantially improves target accuracy before the performance saturates at moderate orders. In contrast, most MAG tasks are less sensitive to changes in k. The filtering coefficient γ produces comparatively small fluctuations across most transfer tasks, indicating that BTGA does not rely on a narrowly tuned filtering trade-off. Overall, although the preferred parameter values vary across source-target pairs, BTGA maintains stable performance within reasonable neighborhoods of the task-specific settings.

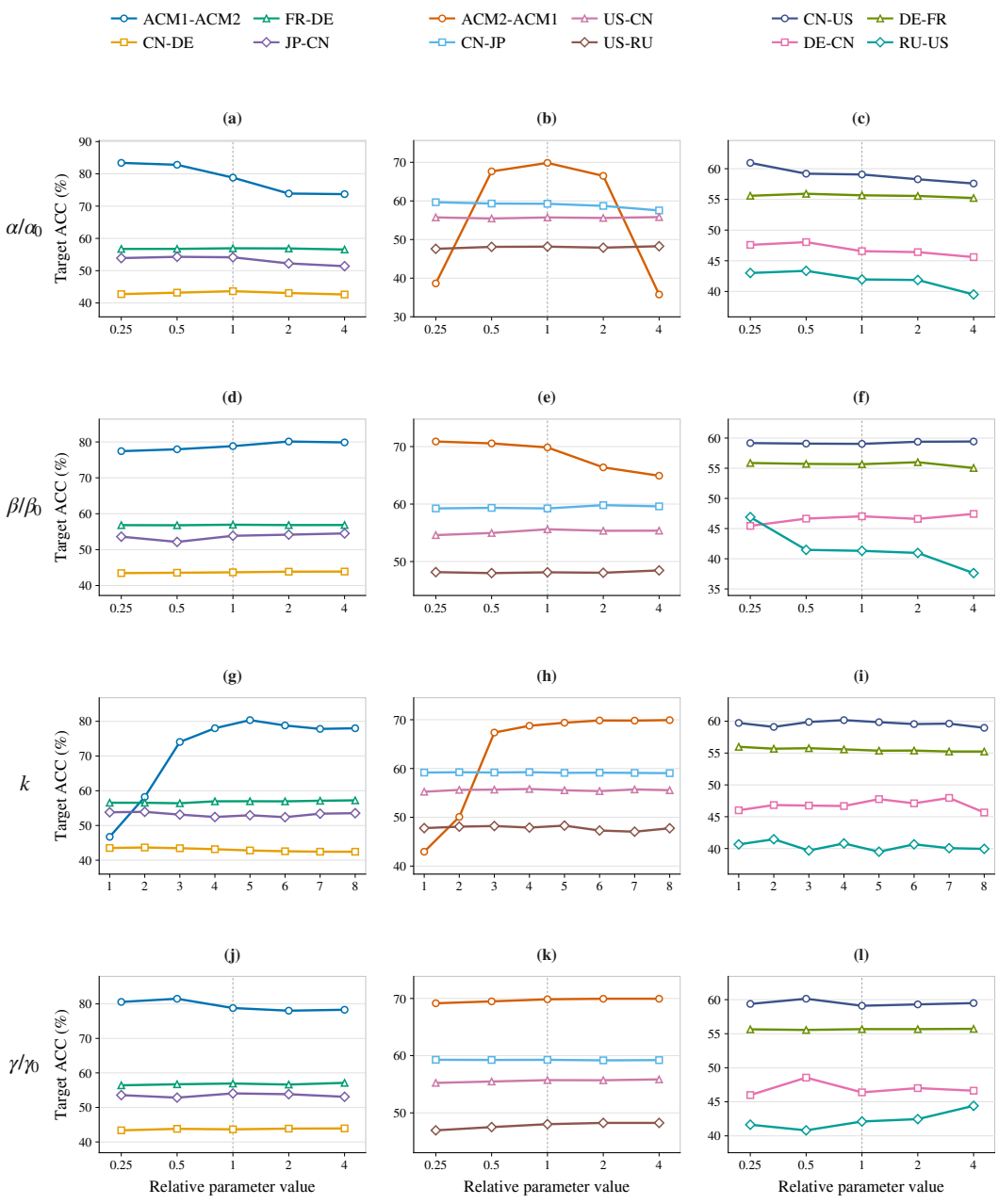


Figure 1: Parameter sensitivity of BTGA over twelve transfer tasks. Rows correspond to α, β, k, and γ, respectively, while columns group different transfer tasks for readability. For α, β, and γ, the horizontal axis denotes the parameter value relative to its task-specific default.

Table 2: Task-specific hyperparameter settings of BTGA.

Transfer task	α	β	k	γ
CN→US	0.5	0.01	2	100
US→CN	5.0	1.00	2	50
JP→CN	2.0	0.05	2	50
CN→JP	5.0	0.10	2	100
DE→FR	5.0	0.10	2	100
FR→DE	0.5	0.10	4	100
RU→US	3.0	0.03	1	100
US→RU	0.5	0.03	2	50
DE→CN	0.5	0.03	2	100
CN→DE	0.5	0.03	2	100
ACM1→ACM2	10.0	0.10	6	100
ACM2→ACM1	10.0	0.50	6	200

C Model Complexity and Efficiency

In this appendix, we further analyze the computational efficiency of BTGA from three aspects: computational complexity, efficient graph filtering, and empirical training time and memory usage.

C.1 Complexity analysis

The computational complexity of BTGA mainly depends on multi-view representation learning, intra-domain feature correlation, and cross-domain alignment. Let V denote the number of relation views, B the batch size, F the input feature dimension, h the hidden dimension, and d the embedding dimension. Multi-view representation learning requires $\mathcal{O}(VB(Fh+hd))$, while intra-domain feature correlation over all unordered view pairs requires $\mathcal{O}\big(V(V-1)Bd^2\big)$. Cross-domain alignment is performed over V^2 source-target view pairs, resulting in $\mathcal{O}\big(V^2B^2d\big)$. Therefore, the overall per-iteration complexity is $\mathcal{O}\big(VB(Fh+hd)+V(V-1)Bd^2+V^2B^2d\big)$. The consistency and redundancy statistics used for adaptive weighting are computed from detached full-domain representations once per epoch and cached for subsequent mini-batches, and therefore require no additional backpropagation through the encoder. In our experiments, all benchmark datasets contain two relation views, resulting in only four source-target view pairs.

C.2 Efficient Computation of Graph Filtering

The graph filtering step involves a node-space matrix inverse, which becomes computationally expensive for large graphs. To improve efficiency, we use the Woodbury matrix identity to transform the inverse into the feature space:

$$\big(X^D(X^D)^\top+\gamma I\big)^{-1}=\frac{1}{\gamma}I-\frac{1}{\gamma^2}X^D\left(I+\frac{1}{\gamma}(X^D)^\top X^D\right)^{-1}(X^D)^\top. \tag{1}$$

Accordingly, the filtered node attributes can be computed directly as

$$\widetilde{X}^{D,v}=\left[\gamma\phi(L^{D,v})X^DX^D\left((X^D)^\top X^D\right)\right]\left((X^D)^\top X^D+\gamma I\right)^{-1}. \tag{2}$$

This reformulation replaces the inversion of an $N_D\times N_D$ matrix with that of an $F\times F$ matrix, where N_D and F denote the numbers of nodes and input features, respectively. Therefore, when $N_D\gg F$, the expensive node-space inversion is avoided. Since graph filtering is independent of the trainable model parameters, the filtered node attributes are computed before iterative optimization and reused during training.

C.3 Empirical efficiency

We further evaluate the empirical efficiency of BTGA against four representative graph domain adaptation methods, including UDAGCN, ASN, SAGA, and GraphATA. The comparison is conducted on four transfer tasks, ACM1→ACM2, ACM2→ACM1, DE→FR, and FR→DE. For a controlled comparison, all methods are trained for 50 epochs using the same data split, random seed, and GPU.

Training time and peak GPU memory are normalized separately for each task by the corresponding BTGA measurement, with BTGA set to 1.0. Target ACC reports the best target accuracy observed within the 50 training epochs.

As shown in Table 3, BTGA achieves a favorable balance between training efficiency and adaptation performance. Its training time remains competitive across all four tasks, while consistently requiring the lowest peak GPU memory. The memory advantage is particularly evident on DE→FR and FR→DE, where SAGA requires more than $5\times$ the peak memory of BTGA and ASN requires more than $14\times$. Meanwhile, BTGA achieves the highest target accuracy on all four transfer tasks, reaching 63.87%, 58.48%, 55.54%, and 55.22%, respectively. These results show that BTGA maintains strong target-domain performance without incurring excessive training time or memory consumption.

Table 3: Comparison of training time, peak GPU memory, and target accuracy on four transfer tasks. Training time and memory are normalized by the corresponding BTGA result for each task (BTGA = 1.0; lower is better), while Target ACC (%) reports the best target accuracy within 50 training epochs.

Transfer task	Method	Time (norm.)	Peak memory (norm.)	Target ACC (%)
ACM1→ACM2	**BTGA**	**1.000**	**1.000**	**63.87**
	UDAGCN	2.040	1.496	23.31
	ASN	1.022	1.407	36.68
	SAGA	0.711	2.698	52.97
	GraphATA	1.440	1.274	10.05
ACM2→ACM1	**BTGA**	**1.000**	**1.000**	**58.48**
	UDAGCN	1.951	1.466	27.90
	ASN	1.025	1.463	31.90
	SAGA	0.800	2.666	35.07
	GraphATA	1.186	1.100	43.34
DE→FR	**BTGA**	**1.000**	**1.000**	**55.54**
	UDAGCN	9.515	1.290	42.60
	ASN	10.072	15.136	20.44
	SAGA	57.818	5.356	53.60
	GraphATA	0.921	2.304	46.79
FR→DE	**BTGA**	**1.000**	**1.000**	**55.22**
	UDAGCN	9.664	1.334	31.52
	ASN	10.095	14.086	29.40
	SAGA	52.578	5.349	53.30
	GraphATA	1.433	4.912	47.38

D Description of Algorithm BTGA

The complete training procedure of BTGA is summarized in Algorithm 1. Graph filtering is first applied to each relation view before model training. The filtered node attributes are then fed into a shared encoder, followed by intra-domain redundancy reduction and adaptive cross-domain alignment. Finally, the model is optimized using the source classification, intra-domain, cross-domain alignment, and target entropy objectives.

E Additional Ablation Study

To further examine the contribution of the main components, we evaluate BTGA on eight additional transfer directions: DE→FR, FR→DE, US→RU, DE→CN, CN→US, US→CN, JP→CN, and CN→JP. We compare the complete model with variants that remove the graph filter (w/o Filter), remove the intra-domain Barlow Twins objective (w/o Intra), remove adaptive cross-domain

Algorithm 1 The proposed BTGA framework

Require: Labeled source graph $\mathcal{G}^S = \{\mathcal{V}^S, \{A^{S,v}\}_{v=1}^V, X^S, Y^S\}$; unlabeled target graph $\mathcal{G}^T = \{\mathcal{V}^T, \{A^{T,v}\}_{v=1}^V, X^T\}$; filter order k; filter coefficient γ; trade-off coefficients α, β; temperature τ; and training epochs T.

Ensure: Target-node predictions $\widehat{Y}^T$.

for each $D \in \{S, T\}$ and relation view $v \in \{1, \ldots, V\}$ **do**
  Construct the self-looped normalized adjacency $\widehat{A}^{D,v}$ and Laplacian $L^{D,v} = I - \widehat{A}^{D,v}$.
  Compute the topology prior $\phi(L^{D,v}) = (I - L^{D,v}/2)^k$.
  Obtain the view-specific filter

$$K^{D,v} = \left[\gamma\phi(L^{D,v}) + X^D(X^D)^\top\right]\left[X^D(X^D)^\top + \gamma I\right]^{-1}.$$

  Generate filtered attributes $\widetilde{X}^{D,v} = K^{D,v}X^D$.
end for
Initialize the shared encoder f_Θ, classifier f_Φ, and learnable coefficient λ_{inter}.
for $t = 1$ to T **do**
  for each $D \in \{S, T\}$ and $v \in \{1, \ldots, V\}$ **do**
    Compute $Z^{D,v} = f_\Theta(\widetilde{X}^{D,v})$.
  end for
  For every unordered view pair (p, q) within each domain, compute the cross-correlation matrix and

$$\mathcal{L}_{\text{BT}}(Z^{D,p}, Z^{D,q}) = \frac{1}{d}\sum_i (1 - M_{ii}^{D,pq})^2 + \frac{1}{d(d-1)}\sum_i\sum_{j\neq i}(M_{ij}^{D,pq})^2.$$

  Average the source- and target-domain pairwise losses to obtain $\mathcal{L}_{\text{intra}}$.
  Concatenate the view representations, project them onto a shared PCA coordinate system, and construct fixed-size positive and negative sigma summaries $H^{D,v}$.
  for each source-target view pair (p, q) **do**
    Standardize $H^{S,p}$ and $H^{T,q}$ and compute $\mathbf{M}^{p,q} = (\widetilde{H}^{S,p})^\top \widetilde{H}^{T,q}/(2r)$.
    Compute $C_{p,q} = 1 - d^{-1}\sum_i (1 - \mathbf{M}_{ii}^{p,q})^2$ and $R_{p,q} = [d(d-1)]^{-1}\sum_i\sum_{j\neq i}(\mathbf{M}_{ij}^{p,q})^2$.
    Compute $s_{p,q} = (1 - C_{p,q}) + \lambda_{\text{inter}} R_{p,q}$.
  end for
  Compute $\omega_{p,q} = \exp(-s_{p,q}/\tau)/\sum_{a=1}^V\sum_{b=1}^V \exp(-s_{a,b}/\tau)$ and $\mathcal{L}_{\text{align}} = \sum_{p=1}^V\sum_{q=1}^V \omega_{p,q}\,\text{MK-MMD}^2(Z^{S,p}, Z^{T,q})$.
  Aggregate view representations and obtain $\widehat{Y}^S$ and $\widehat{Y}^T$ with the shared classifier.
  Compute the source classification loss $\mathcal{L}_S$ and target entropy loss $\mathcal{L}_T$.
  Optimize $\mathcal{L} = \mathcal{L}_S + \mathcal{L}_{\text{intra}} + \alpha\mathcal{L}_{\text{align}} + \beta\mathcal{L}_T$ with respect to Θ, Φ, and λ_{inter}.
  Detach the statistical summaries used for pair weighting and refresh them for the next epoch.
end for
return $\widehat{Y}^T$.

alignment (w/o Alignment), or use only consistency or only redundancy to determine the view-pair weights.

As shown in Figure 2, BTGA (Full) achieves the highest accuracy on all eight transfer directions. Removing the graph filter or the intra-domain objective consistently reduces performance, indicating that topology-aware view representations and cross-view redundancy reduction provide complementary benefits. The largest degradation is observed for the w/o Alignment variant on several tasks, which confirms that the weighted cross-domain alignment is central to transferring information between domains. The consistency-only and redundancy-only variants also remain below the complete model, showing that the two statistics are most effective when used jointly to characterize source-target view pairs.

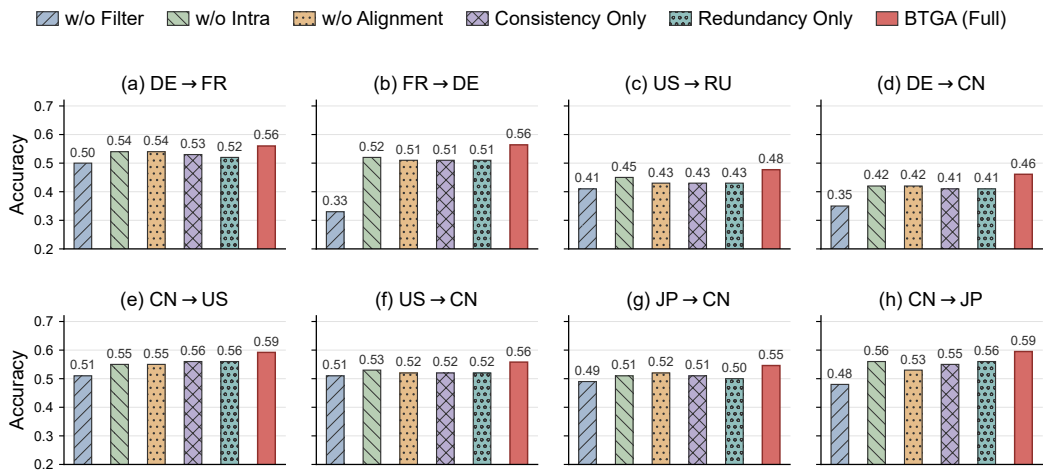


Figure 2: Additional ablation results of BTGA on eight transfer directions. The complete model is compared with variants that remove graph filtering, remove intra-domain redundancy reduction, remove adaptive alignment, or use only consistency or only redundancy for view-pair weighting.

The full model remains the strongest variant across the additional transfer directions. The consistent degradation caused by removing filtering, intra-domain redundancy reduction, or adaptive alignment supports the complementary role of these components, while the single-statistic variants confirm that consistency and redundancy are most effective when used jointly.

To further assess the role of graph filtering, we compare the proposed filter with low-pass filtering, mix-pass filtering, and no filtering on four transfer tasks. Figure 3 reports target accuracy and macro-F1 for these variants. The results show that the effect of filtering depends on the transfer task and metric, while removing filtering can lead to a noticeable performance drop on several tasks.

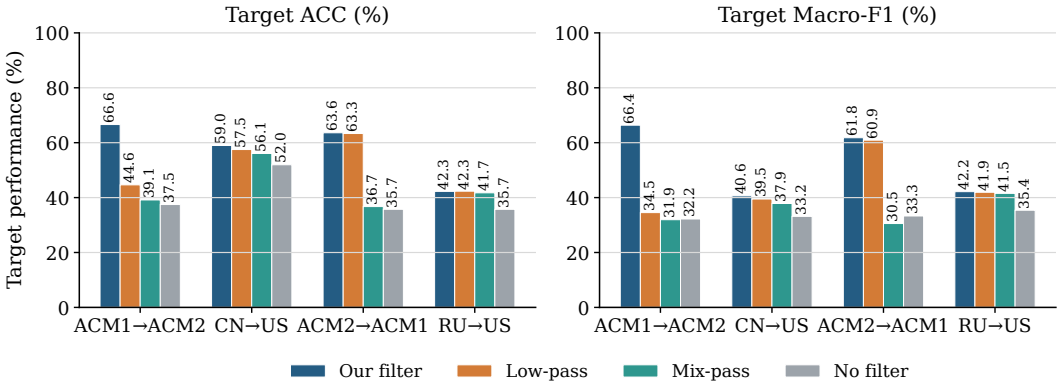


Figure 3: Target accuracy and macro-F1 under different graph-filtering variants on four transfer tasks.

F Additional Visualization Results

To provide an additional qualitative assessment of cross-domain representation alignment, Figure 4 visualizes the node embeddings learned by BTGA on the ACM2→ACM1 transfer task. The visualization shows the source and target embeddings in a shared two-dimensional space. The class clusters are separated while source and target points with the same label remain interleaved, providing qualitative evidence that BTGA preserves class-discriminative structure while reducing the cross-domain discrepancy.



(a) ASN (b) PA (c) UDAGCN (d) SAGA (e) BTGA

Figure 4: t-SNE visualization of source and target node embeddings on the ACM2→ACM1 transfer task.