Target-adaptive Structure-Semantic Consistency for Unsupervised Graph Domain Adaptation

Yan Zou[1], Yongzheng Lu[1], Na Li[1], Xiatian Zhu[2], Lan Du[3], Ming Yan[4], and Ying Ma[1] (✉)

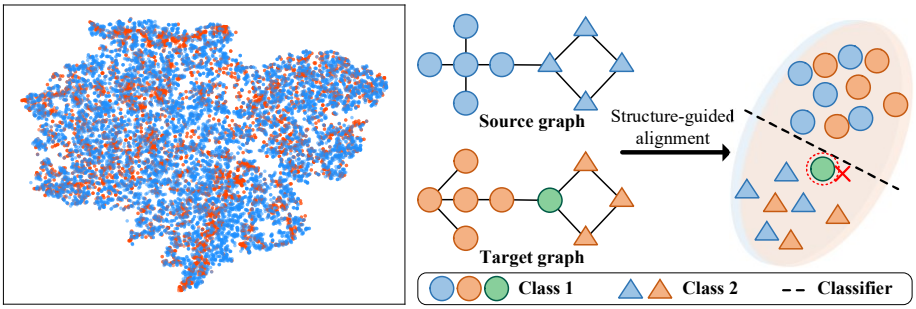
[1] Harbin Institute of Technology, Harbin, China {23b903066, 24b903068, 24s003043}@stu.hit.edu.cn, y.ma@hit.edu.cn
[2] University of Surrey, Guildford, UK xiatian.zhu@surrey.ac.uk
[3] Monash University, Melbourne, VIC, Australia lan.du@monash.edu
[4] Centre for Frontier AI Research (CFAR), A*STAR, Singapore yanmingtop@gmail.com

Abstract. Unsupervised Graph Domain Adaptation (UGDA) aims to mitigate distribution shifts between domains by transferring knowledge from labeled source graphs to unlabeled target graphs. Current work indicates that enhancing target embeddings is helpful for domain generalization. However, these methods primarily focus on structure-guided enhancement but often overlook the intrinsic coupling between structural topology and node semantics in graph data, resulting in suboptimal target representations during complex structure adaptation. To address this problem, we propose a novel approach called Target-adaptive Structure-Semantic Consistency (TASSC). First, we establish bidirectional optimization, ensuring consistency between structural proximity and semantic similarity on the target graph. Specifically, we propose a hybrid contrastive learning strategy, which unifies topological neighbors and cosine-similarity features (semantic neighbors) as positive samples. Additionally, we employ entropy minimization to suppress target semantic ambiguity caused by source domain biases, creating a closed-loop optimization where 'structure guides semantics, semantics feedback structure.' Furthermore, we develop a scale-aware adaptive module to access scale disparities between domains, dynamically transferring source knowledge to mitigate target semantic insufficiency. Extensive experiments on three real-world benchmark datasets demonstrate that our method achieves state-of-the-art results.

Keywords: Graph Neural Networks · Transfer Learning · Unsupervised Graph Domain Adaptation.

1 Introduction

Graph Neural Networks (GNNs) [7,16] have demonstrated remarkable effectiveness in modeling relational data, driving advances in social network analysis [25], protein interaction prediction [37], and recommendation systems [24]. However, their performance heavily relies on labeled data, with cross-domain generalization hindered by label scarcity and distribution shifts. To mitigate such issues,



(a) (b)

Fig. 1. Illustration of motivation. (a) The t-SNE [17] feature visualization for the task A → C on the ArnetMiner [27] dataset, where orange and blue colors represent the target and source data, respectively. (b) The performance of structure-guided alignment is limited without the semantic consistency constraint.

Unsupervised Graph Domain Adaptation (UGDA) [31,8,23,13] addresses this by transferring structural knowledge from labeled source graphs to unlabeled target graphs via domain-invariant representation learning.

In real-world scenarios, such as citation networks across different publishers and time periods, variations in node attributes and graph structural properties are evident. The source graph may contain dense clusters of highly interconnected papers, whereas the target graph may have a more sparse, distributed structure, or vice versa. This drives recent efforts to leverage target information for better domain generalization (DG). A2GNN [11] implicitly enhances topology representation by stacking multiple message-passing layers on the target graph, while TDSS [3] further employs structural smoothing to mitigate distribution shifts. Moreover, some studies [14,15] adjust edge weights to better align target topology, mitigating conditional shifts caused by neighborhood structures.

However, these methods primarily adopt target structure-guided alignment to refine target embeddings while overlooking the importance of target-specific semantic information. As illustrated in Fig. 1(a), in the embedding space, target node representations become diffusely distributed due to insufficient semantic constraints. This results in a situation where the source domain, benefiting from labeled supervision, could partially ignore the semantic constraint to some extent and still achieve good classification performance, as shown in the upper part of Fig. 1(b). In contrast, the target domain lacks such guidance, causing the node (marked by a green circle in Fig. 1(b)) within the same class to experience shifts due to structural perturbations, which may lead to misclassification. Therefore, it is essential to not only align features based on target topology but also ensure consistency between structure and semantics. Specifically, nodes with strong structural connections should remain close in feature space, reflecting the graph topology, while semantically similar nodes should form compact clusters, preserving consistency for accurate classification.

Motivated by this analysis, we propose Target-adaptive Structure-Semantic Consistency (TASSC) for cross-domain node classification. First, we design a structure-semantic constraint module to break through the limitations of single-perspective optimization. Specifically, we introduce a hybrid contrastive learning strategy, constructing positive and negative sample pairs at both the node and embedding levels to enforce dual consistency. At the node level, positives are defined by topological neighbors, while at the embedding level, they are determined through cosine similarity. Additionally, we incorporate entropy minimization on the target graph to reduce noise from source structural priors, enhancing semantic consistency. Second, we develop an adaptive adjustment module to assess the target domain's information scale relative to the source domain. By selectively transferring high-quality source knowledge, this module adaptively enhances target semantics. These two modules are designed to preserve structure-semantic consistency during adaptation, refining target node representations through topological properties and semantic similarity, ultimately improving generalization on the target graph.

Our contributions can be summarized as follows:

1. We propose TASSC, a novel framework for UGDA, which establishes a bi-directional adaptation process, integrating structural topology and semantic representations on the target graph to enhance representation consistency and improve cross-domain generalization.
2. We design an adaptive mechanism that assesses domain complexity and refines target representations by dynamically leveraging source domain information, facilitating more effective knowledge transfer.
3. We conduct extensive experiments on three benchmark datasets. The results demonstrate that TASSC achieves state-of-the-art performance, significantly outperforming the best baseline with notable relative improvements.

2 Related work

Unsupervised Domain Adaptation (UDA) [5,6] is a widely adopted transfer learning paradigm aimed at minimizing domain divergence. In the context of graph-structured data, UGDA [4,23] has emerged as an effective approach for addressing distribution shifts within relational networks. Existing UGDA methods can be categorized into three main groups.

Methods in the first group primarily focus on reducing source prediction risks through enhanced node embeddings [4,33]. For example, UDAGCN [31] introduces a dual-GNN that employs adversarial training to align feature distributions across domains. ACDNE [21] utilizes the k-hop PPMI matrix to capture high-order proximity, ensuring global consistency during adaptation. The second group of methods mitigate graph domain shifts by constructing intermediate representations to bridge domain gaps effectively [34,30]. Several strategies have been proposed to achieve this. ASN [36] separates domain-private and shared features, leveraging adversarial domain adaptation to extract the domain-invariant

shared features across networks. GGDA [10] introduces a compact domain sequence with FGW-based intermediate graphs and vertex-based progression to minimize information loss and improve adaptation. KBL [1] adds data augmentation with adversarial learning to align node embeddings across domains. The last group of methods enhances target embeddings by leveraging target domain information [28,14,12]. PairAlign [15] mitigates structure and label shifts by adjusting node influences with edge and label weights, improving cross-domain alignment in graph adaptation. DMGNN [22] refines predictions through a label propagation mechanism, utilizing a node classifier that improves label consistency by aggregating the predictions of a node and its neighbors. A2GNN [11] enhances adaptation by replacing a shared encoder with a shared transformation layer and deeper propagation on the target graph. TDSS [3] mitigates structural shifts and improves node representation by performing structural smoothing on the target graph.

While these methods above have shown remarkable effectiveness in enhancing target node representations, the structure-semantic collaborative mapping mechanism has not received sufficient attention and exploitation.

3 Preliminaries

In this section, we begin with the notation and problem definition, then present the graph neural networks employed in this work.

3.1 Notation and problem definition

Consider an attributed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_i\}_{i=1}^{|\mathcal{V}|}$ is the set of $|\mathcal{V}|$ nodes and $\mathcal{E}$ is the set of edges. Let $\mathbf{X} = \{\mathbf{x}_v | v \in \mathcal{V}\} \in \mathbb{R}^{|\mathcal{V}| \times D}$ be the node adjacency matrix of $\mathcal{G}$, where $\mathbf{x}_i$ is the D-dimensional feature vector of node v_i. We denote the graph structure as an adjacency matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$, where $\mathbf{A}_{i,j} = 1$ means there exits an edge connecting v_i and v_j, otherwise $\mathrm{A}_{i,j} = 0$.

Given two different but related source domain $\mathcal{S}$ and target domain $\mathcal{T}$, both domains share the same C categories. The source graph with m labeled nodes is denoted as $\mathcal{G}_S = (\mathcal{V}_S, \mathcal{E}_S, \mathcal{Y}_S)$ and target graph with n unlabeled nodes is denoted as $\mathcal{G}_T = (\mathcal{V}_T, \mathcal{E}_T)$. In practice, $\mathbb{P}(\mathcal{G}_S) \neq \mathbb{P}(\mathcal{G}_t)$ but $\mathbb{P}(\mathcal{Y}_S | (\mathcal{V}_S, \mathcal{E}_S)) = \mathbb{P}(\mathcal{Y}_T | (\mathcal{V}_T, \mathcal{E}_T))$, where $\mathbb{P}$ denotes the probability distributions. Our goal is to build a function $\mathcal{F} : (\mathcal{G}_S, \mathcal{G}_T) \rightarrow \mathcal{Y}$ for the node classification task, using the labeled source graph $\mathcal{G}_s$ and unlabeled target graph $\mathcal{G}_T$ during adaptation.

3.2 Graph neural networks.

Inspired by the previous work [11], we construct our model $\mathcal{F}$, which adopts an asymmetric strategy that increases the number of message passing layers on the target graph while reducing those on the source graph. Specifically, the feed-forward neural network $\mathcal{F}$ with parameter Θ consists of two modules: a feature encoder $g(\cdot;\theta) : \mathbf{X} \rightarrow \mathbb{R}^d$ and a node classifier $h(\cdot;\xi) : \mathbb{R}^d \rightarrow \mathbb{R}^C$, i.e.

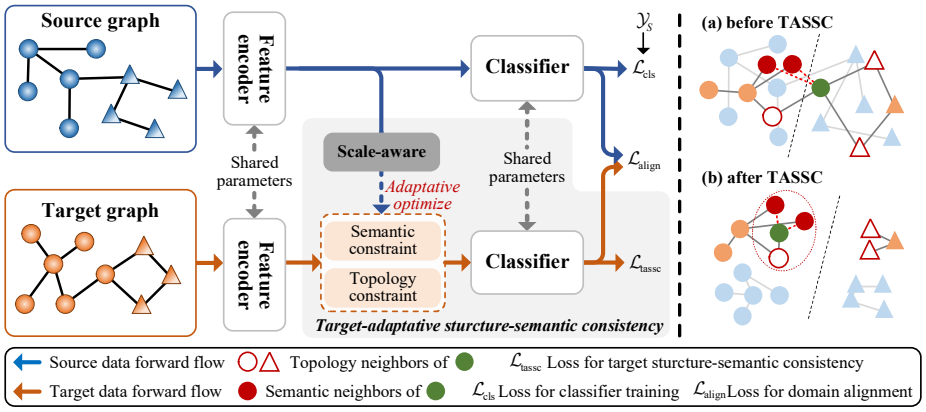


Fig. 2. The framework of the proposed method to learn $\mathcal{F}$.

$\mathcal{F} = h \circ g(\cdot; \Theta)$ where $\Theta = \{\theta, \xi\}$ represents the shared parameters across both the source and target graphs, and d denotes the dimension of the input features.

Without loss of generality, we update the representation of node v at l-th hidden layer in the network $\mathcal{F}$, which can be formulated as:

$$\mathbf{h}_v^l = \mathcal{C}^l\left(\mathbf{h}_v^{l-1}, \mathcal{A}^{(k)}\left(\left\{h_u^{l-1} | u \in \mathcal{N}(v)\right\}\right)\right), \tag{1}$$

where $\mathbf{h}_v^l = \mathcal{F}^l(v; \Theta)$, $\mathcal{N}(v)$ represents the neighbors of node v and $\mathcal{C}^l$ serves as the combination function at layer l. The aggregation function $\mathcal{A}^{(k)}$ comprehensively processes the features of the k-hop neighbors of v. Notably, the value of node k in the source graph branch is significantly smaller than in the target graph branch, i.e., $k_S \ll k_T$.

4 Methodology

In this section, we introduce our method, Target-adaptive Structure-Semantic Consistency (TASSC). The overall framework of TASSC is shown in Fig. 2. In the following sections, we provide a detailed explanation of its components.

4.1 Target structure-semantic consistency learning

The core contribution of this paper is to efficiently preserve the consistency between target structural proximity and semantic similarity, to achieve better generalization. Notably, we observe that graph contrastive learning [24,32] provides distinct theoretical advantages. By explicitly defining structure-guided similarity relationships, this approach inherently incorporates local topological constraints into the representation learning process. For instance, by forming positive pairs within local neighborhoods, the contrastive loss encourages the latent space distance between adjacent nodes to be proportional to their structural similarity,

thereby reinforcing the alignment between the learned representations and the underlying graph structure.

However, relying solely on topological neighbors as positive samples may propagate structural noise, especially under cross-domain distribution shifts. As shown on the right side of Fig. 2, the green-circled node is connected to three other nodes, but only one of them shares the same class. This could lead to misclassification due to source-biased structural alignment, affected by the hollow red-triangled nodes. In contrast, the solid red-circled nodes, though not directly connected to the green-circled node (as shown by the red dashed line in Fig. 2), are semantically similar to it, as indicated by their proximity in the feature space. Therefore, the solid red-circled nodes can provide meaningful semantic information to the green-circled node, effectively acting as its semantic neighbors.

Motivated by this observation, we propose a hybrid positive sampling strategy for target-oriented contrastive learning. In this approach, positive pairs are constructed by combining both topological neighbors and feature-similar nodes. For any input target node v_t, we calculate its cosine similarity with all other nodes in the target graph $\mathcal{G}_t$, excluding v_t itself. Let $\hat{\mathcal{V}}_t \subset \mathcal{V}_t$ denote the set of nodes used for comparison, where $v_t \notin \hat{\mathcal{V}}_t$. Thus, we select semantic neighbors based on similarity ranking, which can be formulated as:

$$\mathcal{N}_{\text{sem}}(v_t) = \left\{ v_{t,i}^{\text{sem}} \mid v_{t,i}^{\text{sem}} \in \hat{\mathcal{V}}_t, i \in \text{topk}\left(\text{sim}\left(\mathbf{v}_t, \hat{\mathbf{V}}_t\right), k_{\text{sem}}\right) \right\}, \tag{2}$$

where $\mathbf{v}_t = g(v_t; \theta)$ presents the target feature of v_t, $\hat{\mathbf{V}}_t = [\hat{\mathbf{v}}_{t,j}]_{j=1}^{|\mathcal{V}_t|-1}$ is the matrix containing the node feature $\hat{\mathbf{v}}_t = g(\hat{v}_t; \theta)$ for each node $\hat{v}_t \in \hat{\mathcal{V}}_t$. Additionally, $\text{sim}(\cdot, \cdot)$ is the function to evaluate cosine-similarity of the input two vectors, and $\text{topk}(\mathcal{Z}, k)$ represent an operation which selects the k elements with the highest values from set $\mathcal{Z}$ and returns the corresponding indices of these elements.

For the one-hop local neighbor set $\mathcal{N}(v_t)$, we uniformly select k_{topo} elements as the topological neighbors of v_t, forming the set $\mathcal{N}_{\text{topo}}(v_t)$ [32]. We then dynamically select positive samples for v_t as follows:

$$\mathcal{N}_+(v_t) = \mathcal{N}_{\text{topo}}(v_t) \cup \mathcal{N}_{\text{sem}}(v_t), \tag{3}$$

where $\mathcal{N}_+(v_t)$ denotes the set of selected target neighbors for v_t.

For any input target node v_t, the regularization can be expressed as:

$$\mathcal{L}_{\text{ssc}}^{\mathcal{T}}(\mathcal{G}_T; \Theta) = -\mathbb{E}_{v_t \in \mathcal{V}_T} \mathbb{E}_{v_{t+} \in \mathcal{N}_+(v_t)} \log \frac{e^{\mathbf{v}_t^\top \mathbf{v}_{t+}/\tau}}{e^{\mathbf{v}_t^\top \mathbf{v}_{t+}/\tau} + \sum_{v_{t-} \in \mathcal{N}_-(v_t)} e^{\mathbf{v}_t^\top \mathbf{v}_{t-}/\tau}}. \tag{4}$$

Here $\mathbf{v}_{t+}$ and $\mathbf{v}_{t-}$ are the node representations generated by $g(\cdot; \theta)$ for nodes v_{t+} and v_{t-}, respectively. τ is the temperature hyper-parameter. Typically, $\mathcal{N}_-(v_t)$ denotes the negative sample set, which is randomly selected from the target node space $\mathcal{V}_T$ [26], with the constraint that the number of negative samples, $|\mathcal{N}_-(v_t)|$, is equal to the number of positive samples, $|\mathcal{N}_+(v_t)|$, ensuring equal sample sizes for effective contrastive learning.

4.2 Target-oriented semantic regularization

While the target structure-semantic consistency learning method proposed in Section 4.1 effectively enforces localized constraints between topological neighborhoods and semantic neighborhoods, it lacks explicit regularization of the global semantic distribution in the embedding space. Specifically, under the shared feature encoder framework, domain structural biases may cause target node representations to deviate from their true semantic distribution, resulting in blurred class boundaries.

To address this limitation, incorporating global semantic constraints or regularization mechanisms into the learning objective is essential. Such enhancements would ensure that target representations not only preserve local consistency but also maintain global discriminability, thereby mitigating the adverse effects of noise introduced by source structural priors.

Thus, we employ entropy minimization to preserve critical distinctions in the target graph's feature space, formulating this regularization as shown in Eq.(5).

$$\mathcal{L}_{\text{ent}}^{\mathcal{T}}\left(\mathcal{G}_T;\Theta\right)=-\mathbb{E}_{v_t\in\mathcal{V}_T}\sum_{c=1}^{C}\delta_c\left(\mathcal{F}\left(v_t;\Theta\right)\right)\log\delta_c\left(\mathcal{F}\left(v_t;\Theta\right)\right), \tag{5}$$

where $\delta_c\left(\boldsymbol{\mu}\right)=\exp\left(\boldsymbol{\mu}_c\right)/\sum_i\exp\left(\boldsymbol{\mu}_i\right)$ denotes the c-th element in the *softmax* output for a C-dimensional vector $\boldsymbol{\mu}$, and $\mathcal{F}\left(v_t;\Theta\right)=h\circ g\left(v_t;\{\theta,\phi\}\right)$ denotes the C-dimensional output of target node v_t.

Due to the absence of target labels, the model may overfit to dominant classes under the optimization of Eq. (5), causing minority classes to collapse into ambiguous clusters. To tackle this issue, we integrate a diversity maximization regularization term to counteract degenerate distributions [9], which can be formulated as:

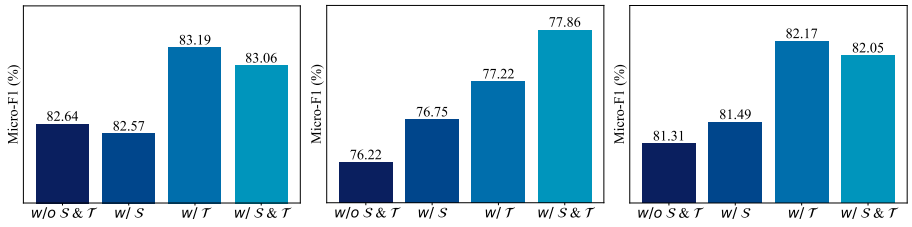
$$\mathcal{L}_{\text{bal}}^{\mathcal{T}}\left(\mathcal{G}_T;\Theta\right)=\text{KL}\left(\boldsymbol{p},\frac{1}{C}\mathbb{I}_C\right), \tag{6}$$

where $\boldsymbol{p}=\mathbb{E}_{v_t\in\mathcal{V}_T}[\delta\left(\mathcal{F}\left(v_t;\Theta\right)\right)]$ represents the mean output embedding of the whole target domain, $\mathbb{I}_C$ is a C-dimensional vector with all elements set to one. $\text{KL}\left(\cdot,\cdot\right)$ is a function outputting Kullback–Leibler(KL) divergence of the two inputted probability vectors.

Combining Eq.(5) and Eq.(6), we concisely express the target-oriented semantic regularization by Eq.(7).

$$\mathcal{L}_{\text{tos}}^{\mathcal{T}}\left(\mathcal{G}_T;\Theta\right)=\mathcal{L}_{\text{ent}}^{\mathcal{T}}\left(\mathcal{G}_T;\Theta\right)+\mathcal{L}_{\text{bal}}^{\mathcal{T}}\left(\mathcal{G}_T;\Theta\right). \tag{7}$$

By maximizing mutual information between target representations and latent labels, $\mathcal{L}_{\text{tos}}^{\mathcal{T}}$ drives the model to learn more distinct and discriminative target node representations in the latent space.



(a) A → C (b) C → D (c) D → C

Fig. 3. The Micro-F1 scores (%) of different strategies for achieving structural and semantic consistency on the ArnetMiner dataset.

4.3 Scale-aware adaptive structure-semantic consistency learning

Similar to previous works [11], our method mitigates distribution shifts under the assumption of implicit homogeneity between source and target domain scales. However, we observe that this assumption overlooks the dynamic impact of scale discrepancies on semantic optimization, potentially resulting in suboptimal class separability and ambiguous node representations in the target graph.

To validate our hypothesis, we conduct experiments on the widely used ArnetMiner [27] dataset in UGDA for further investigation. Fig. 3 shows the results under different strategies. Specifically, we design four distinct experiments on three tasks, where $w/$ / w/o $\mathcal{S}$ or $\mathcal{T}$ indicates whether employ structure-semantic consistency learning as Eq. (4), is applied to the source or target domain data. As shown in Tab. 1 for the detailed dataset, domains A and C exhibit similar numbers of nodes and edges, whereas domain D is relatively smaller in scale. When we transfer knowledge from one domain to another which containing similar or larger scale information (e.g. A → C, D → C), as shown in Fig. 3(a) and Fig. 3(c), the source semantics have little impact on overall generalization. Moreover, integrating source semantics with target information may even lead to performance degradation. In contrast, enforcing both structural and semantic constraints solely on the target data significantly improves generalization. The results are explainable that when the target graph carries information on par with or exceeding that of the source graph, further enhancing the source representation introduces additional structural noise, thereby compromising target semantics. Nonetheless, as illustrated in Fig. 3(b), transferring knowledge to a smaller-scale domain in the C → D task results in insufficient informational support, leading to suboptimal performance. However, incorporating semantic optimization from the source domain can enhance cross-domain semantic commonality, thereby improving target representation.

The scale of a graph is determined by both the number of nodes and edges. Nodes serve as the main carriers of semantic information in graph-structured data, while edges serve as concrete representations of structural relationships. Guided by the aforementioned analysis, we design the weighted joint scale dis-

parity measurement method, which is formulated as follows:

$$\lambda = \begin{cases} 1, & if \quad \omega \frac{|\mathcal{V}_S|}{|\mathcal{V}_T|} + (1-\omega)\frac{|\mathcal{E}_S|}{|\mathcal{E}_T|} \geq \epsilon \ \wedge \ \frac{|\mathcal{E}_T|}{|\mathcal{V}_T|} \leq \Upsilon \\ 0, & otherwise \end{cases}, \tag{8}$$

where the weight coefficient $\omega = 0.7$ is used to balance the impact of nodes and edges on domain-scale measurement. ϵ is the value that represents the relative information scale of the source domain compared to the target domain and is set to 1.5. Additionally, we impose a constraint ensuring that the target degree, defined as $\frac{|\mathcal{E}_T|}{|\mathcal{V}_T|}$, remains below the threshold Υ, thereby preventing the model from neglecting extreme structural variations.

Similar to Eq. (4), we build the structure-semantic consistency learning on the source domain, which is regularized as follows, where $\mathbf{v}_s = g(v_s; \theta)$:

$$\mathcal{L}^{\mathcal{S}}_{\text{ssc}}(\mathcal{G}_S; \Theta) = -\mathbb{E}_{v_s \in \mathcal{V}_S} \mathbb{E}_{v_{s+} \in \mathcal{N}_+(v_s)} \log \frac{e^{\mathbf{v}_s^\top \mathbf{v}_{s+}/\tau}}{e^{\mathbf{v}_s^\top \mathbf{v}_{s+}/\tau} + \sum_{v_{s-} \in \mathcal{N}_-(v_s)} e^{\mathbf{v}_s^\top \mathbf{v}_{s-}/\tau}}. \tag{9}$$

By incorporating the scale-adaptive adjustment parameter λ derived from Eq. (8), we implement Eq. 10, where α and β are trade-off parameters.

$$\mathcal{L}_{\text{tassc}}(\mathcal{G}_S, \mathcal{G}_T; \Theta) = \alpha \left(\mathcal{L}^{\mathcal{T}}_{\text{ssc}}(\mathcal{G}_T; \Theta) + \lambda \mathcal{L}^{\mathcal{S}}_{\text{ssc}}(\mathcal{G}_S; \Theta) \right) + \beta \mathcal{L}^{\mathcal{T}}_{\text{tos}}(\mathcal{G}_T; \Theta). \tag{10}$$

4.4 Model optimization

As aforementioned, we achieve the target-adaptive structure-semantic consistency learning by minimizing the objective $\mathcal{L}_{\text{tassc}}$. To facilitate effective knowledge transfer from the source domain to the target domain, we combine the standard cross-entropy loss and entropy minimization loss on the source domain. The resulting regularization is formulated as shown in Eq.(11).

$$\begin{aligned} \mathcal{L}_{\text{cls}}(\mathcal{G}_S; \Theta) = &- \mathbb{E}_{(v_s, y_s) \in \mathcal{V}_S \times \mathcal{Y}_S} \sum_{c=1}^{C} q_c \log \delta_c \left(\mathcal{F}(v_s; \Theta) \right) \\ &- \gamma_1 \mathbb{E}_{v_s \in \mathcal{V}_S} \sum_{c=1}^{C} \delta_c \left(\mathcal{F}(v_s; \Theta) \right) \log \delta_c \left(\mathcal{F}(v_s; \Theta) \right), \end{aligned} \tag{11}$$

where $\gamma_1 = 0.5$, and q denotes the one-hot encoding of $y_s \in \mathcal{Y}_S$, with q_c being '1' for the correct class and '0' for all other classes.

To summarize, the overall loss function can be expressed as:

$$\mathcal{L}(\mathcal{G}_S, \mathcal{G}_T; \Theta) = \mathcal{L}_{\text{tassc}}(\mathcal{G}_S, \mathcal{G}_T; \Theta) + \mathcal{L}_{\text{cls}}(\mathcal{G}_S; \Theta) + \gamma_2 \mathcal{L}_{\text{align}}(\mathcal{G}_S, \mathcal{G}_T; \Theta), \tag{12}$$

where $\mathcal{L}_{\text{align}}$ is the domain alignment regularization, γ_2 functions as a weighting factor. In this paper, we follow the recent work [3] which utilizes the Maximum Mean Discrepancy (MMD) to align the source and target domains.

Table 1. Statistics of datasets, # means 'number of', 'Attr.' refers to attributes.

Datasets	Graph	#Node	#Edge	#Attr.	#Label
ArnetMiner	ACMv9(A)	9,360	15,556	6,775	5
	Citationv1(C)	8,935	15,098		
	DBLPv7(D)	5,484	8,117		
Airport	EUROPE(E)	399	11,990	241	4
	USA(U)	1,190	27,198		
Twitch	Germany(DE)	9,498	153,138	3,170	2
	England(EN)	7,126	35,324		

5 Experiments

This section briefly introduces the evaluation datasets and experimental setting, followed by a presentation of experimental results of our method. Subsequently, supporting experiments, such as analysis and ablation study, are conducted.

5.1 Experimental setup

Datasets. This paper uses the following three datasets that are widely used for UGDA. **ArnetMiner** [27] is a middle-scale dataset, which includes papers from different sources and time periods of 5 categories shared by three domains: ACMv9(A), Citationv1(C), and DSLR(D). Our method is evaluated by performing domain adaptation on all 6 tasks. **Airport** [19] is a small-scale dataset consisting of 4 node categories in total, where each node indicates an airport and each edge represents the routes between two airports. We evaluate our method on two graphs in this dataset: EUROPE(E) and USA(U). **Twitch** [20] is a social network dataset collected from different regions, where each node represents a user and the edges indicate friendships between users. We evaluate our method on the following two graphs: Germany(DE) and England(EN). The statistics of these datasets are presented in Table 1.

Implementation Details. In practice, we adopt the same experimental settings as in previous work [11,3], with slight difference. We design the feature encoder and classifier, respectively, of 128 (except 64 for Twitch) and C units, in which C differs from one dataset to another. Υ is set to 20. The number of topology and semantic neighbors $|\mathcal{N}_+(v_t)| = 5$, where $k_{\text{sem}} = 1$. β in Eq. (10) is initialized at 0.3 starting from the middle stage of training, with the remaining periods set to 0. α and τ are searched within the sets $\{0.5, 1.0\}$ and $\{0.25, 0.5, 1.0\}$, respectively. γ_2 in Eq. (12) is set to 10. We set the message propagation layers $k \in \{1, 10\}$ in the target graph branch while $k = 0$ in the source graph branch. All experiments are implemented using PyTorch and run on a single NVIDIA RTX 3090 with 24GB. The whole network is trained by Adam with weight decay $\in \{1e{-}3, 5e{-}3\}$ and the learning rate is set to 0.01. Under random seed 200, we run the codes repeatedly for 5 rounds and report the average results using both Micro-F1 and Macro-F1 scores. Our code is available on `https://github.com/YeewZ/TASSC`.

Table 2. Classification accuracy (%) on ArnetMiner dataset, the bold and the underline mean the best and the second-best result, respectively.

Methods	A → C		A → D		C → A		C → D		D → A		D → C		Avg.	
	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
UDAGCN [31]	72.15	60.33	66.95	64.83	66.80	67.22	71.77	69.46	58.16	55.89	73.28	61.12	68.19	63.14
SAGDA [18]	77.50	74.09	70.46	66.28	69.90	68.89	73.80	68.10	61.74	53.62	73.92	70.38	71.22	66.89
AdaGCN [4]	79.32	76.51	75.04	71.39	71.67	70.77	75.59	72.34	69.67	69.47	78.20	74.22	74.92	72.45
CWGCN [29]	80.21	78.34	74.11	71.84	71.68	71.80	76.40	73.76	68.35	68.39	76.82	73.73	74.60	72.98
ACDNE [21]	81.75	80.09	76.24	73.59	73.59	74.79	77.21	75.74	71.29	72.64	80.14	78.83	76.70	75.95
GRADE [30]	76.04	72.52	68.22	63.03	69.55	69.34	73.95	70.02	63.72	59.35	74.32	69.32	70.97	67.26
KBL [1]	77.66	75.24	69.60	65.80	70.59	69.87	74.48	70.95	63.23	57.51	74.93	70.28	71.75	68.28
ASN [36]	80.64	77.81	73.80	71.40	72.74	73.17	76.36	73.98	70.15	71.49	78.23	75.17	75.32	73.84
DMGNN [22]	81.58	80.08	76.81	74.76	72.70	73.82	76.57	74.08	70.50	71.44	80.26	78.16	76.40	75.39
GGDA [10]	81.90	80.40	77.20	74.90	**77.10**	76.60	77.40	75.90	**75.80**	**75.80**	81.50	80.30	78.48	77.32
StruRW [14]	77.35	72.07	69.10	62.51	67.81	59.77	73.81	66.89	63.27	53.82	72.41	62.94	70.63	63.00
PairAlign [15]	70.88	67.88	65.91	62.35	65.85	65.09	71.04	67.56	59.34	58.77	67.07	64.61	66.68	64.38
SpecReg [34]	80.55	78.83	75.93	73.98	72.04	73.15	75.74	73.64	71.01	72.34	79.04	77.78	75.72	74.95
A2GNN [11]	82.39	81.06	77.14	75.01	74.30	75.74	77.30	74.97	71.79	72.84	80.63	78.11	77.26	76.29
SEPA [12]	82.46	81.11	76.05	74.78	73.88	75.29	78.08	76.97	73.83	74.85	82.82	81.74	77.79	77.39
TDSS [3]	82.51	80.93	**78.14**	74.94	74.63	76.04	78.21	74.78	73.93	75.39	81.12	78.87	78.09	76.83
TASSC (*ours*)	**83.95**	**82.94**	77.91	**76.11**	75.41	**77.11**	**78.40**	**77.23**	74.16	75.68	**83.19**	**82.11**	**78.84**	**78.53**

Baseline methods. To evaluate our method, we select 16 baseline methods divided into the following three groups. The first group includes 5 deep node embedding methods: ACDNE [21], UDAGCN [31], AdaGCN [4], SAGDA [18], and CWGCN [29]. The second group contains 5 methods using GNNs to aggregate topology and node attributes across source and target graphs to address distribution shifts: ASN [36], KBL [1], GRADE [30], DMGNN [22], and GGDA [10]. The third group includes 6 methods that leverage target information to enhance generalization: SpecReg [34], StruRW [14], PairAlign [15], A2GNN [11], SEPA [12], and TDSS [3].

5.2 Experimental results

The node classification accuracies of TASSC and these comparison methods on the three benchmarks are presented from Tab. 2 to Tab. 4. For the dataset ArnetMiner (Tab. 2), TASSC achieves competitive results. In all comparison tasks, TASSC obtains the best result on A → C, C → D, and D → C. Although GGDA performs best on C → A and D → A, its performance exhibits significant fluctuations on the others. Compared to other UGDA methods, our method achieves the best result in average accuracy and surpasses the second-best method GGDA by 0.36% in Micro-F1 and almost 1.21% in Macro-F1 on average. As shown in Tab. 3, TASSC further defeats other methods on Airport. Specially, our method reaches the best accuracy of 55.28% and 53.72% in Micro-F1 and Macro-F1 on average, and also obtains the best on both two tasks. Upon closer inspection of the results on Twitch (Tab. 4), TASSC demonstrates either the best or second-best performance. Specifically, our method achieves the highest performance on the EN → DE task and surpasses the KBL by 8.35% in the Macro-F1 score. Al-

Table 3. Classification accuracy (%) on Airport dataset, the bold and the underline mean the best and the second-best result, respectively.

Methods	E → U		U → E		Avg.	
	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
CWGCN [29]	44.96	39.68	40.60	34.17	42.78	36.93
SAGDA [18]	36.30	28.18	37.09	36.10	36.70	32.14
ACDNE [21]	<u>50.50</u>	<u>48.15</u>	47.37	44.94	48.94	46.55
AdaGCN [4]	46.89	43.56	49.87	47.67	48.38	45.62
ASN [36]	46.64	43.29	42.11	38.31	44.38	40.80
KBL [1]	45.46	35.61	31.83	23.45	38.65	29.53
GRADE [30]	49.83	47.36	48.37	47.21	49.10	47.29
DGDA [2]	42.27	35.59	43.11	36.47	42.69	36.03
StruRW [14]	43.70	41.27	41.10	37.39	42.40	39.33
PairAlign [15]	39.08	36.55	38.60	35.31	38.84	35.93
A2GNN [11]	47.78	45.32	<u>55.39</u>	<u>52.70</u>	<u>51.59</u>	<u>49.01</u>
TDSS [3]	40.08	30.30	53.13	47.93	46.61	39.12
TASSC (*ours*)	**53.51**	**51.17**	**57.04**	**56.27**	**55.28**	**53.72**

though KBL performs outstandingly on the DE → EN task in Twitch, it exhibits mediocre performance on the other tasks. In contrast, TASSC demonstrates the best or second-best performance among the three datasets.

5.3 Analysis

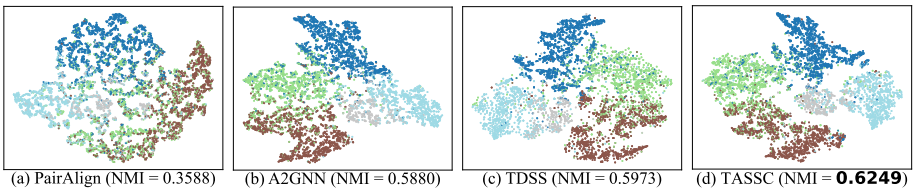
In this section, we further analyze TASSC from the following three aspects.

Feature visualization. According to the 5-way node classification results for the A → C task, we provide the visualization of the node feature representations learned in the target domain by using t-SNE [17], comparing our method against three baselines. As shown in Fig. 4(d), TASSC effectively captures the intrinsic semantic structure of the target graph, yielding a well-clustered distribution. To quantitatively evaluate the clustering quality, we employ Normalized Mutual Information (NMI)[35], and our method achieves the highest score of 0.6249, indicating well-separated clusters and tighter intra-class groupings. In contrast, PairAlign, A2GNN, and TDSS exhibit more dispersion and overlap, with NMIs of 0.3588, 0.5880, and 0.5973, respectively, as shown in Fig. 4(a)-(c).

Parameters sensitivity. In the objective of TASSC, two key parameters, α and β, are introduced, as defined in Eq. (10). Parameter α reflects the adjustment from the contrastive learning based on topology and semantic neighbors. Parameter β regulates the strength of the target semantic regularization. To evaluate their sensitivity, we carry out 60 experiments on the A → C task in the ArnetMiner dataset, varying α within $[0.1, 1.0]$ and β within $[0.05, 0.5]$. As illustrated in Fig. 5, the high-accuracy region (marked in blue) is not isolated, indicating that TASSC is not overly sensitive to specific choices of α and β.

Table 4. Classification accuracy (%) on Twitch dataset, the bold and the underline mean the best and the second-best result, respectively.

Methods	DE → EN		EN → DE		Avg.	
	Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
ACDNE [21]	56.13	55.85	57.30	55.47	56.72	55.66
AdaGCN [4]	57.20	57.20	61.46	58.27	59.33	57.74
CWGCN [29]	57.03	55.00	62.16	55.73	59.60	55.37
UDAGCN [31]	58.32	52.81	62.68	54.04	60.50	53.43
ASN [36]	55.82	55.47	60.45	57.19	58.14	56.33
KBL [1]	**59.32**	**59.11**	<u>64.07</u>	53.90	**61.70**	56.51
GRADE [30]	58.03	57.37	58.71	56.23	58.37	56.80
DMGNN [22]	57.79	53.71	60.24	59.57	59.02	56.64
PairAlign [15]	56.33	55.77	58.28	55.16	57.31	55.47
A2GNN [11]	57.26	56.57	62.44	<u>60.32</u>	59.85	<u>58.45</u>
SEPA [12]	57.90	<u>57.44</u>	63.58	58.34	60.74	57.89
TDSS [3]	56.03	54.88	61.14	55.55	58.59	55.22
TASSC (*ours*)	<u>58.74</u>	57.08	**64.38**	**62.25**	<u>61.56</u>	**59.67**



(a) PairAlign (NMI = 0.3588) (b) A2GNN (NMI = 0.5880) (c) TDSS (NMI = 0.5973) (d) TASSC (NMI = **0.6249**)

Fig. 4. Visualization of the learned node embeddings for the A → C task on the ArnetMiner dataset across different models.

Qualitative study. When selecting semantic neighbors, we fix $k_{\text{sem}} = 1$ without further adjustment. In this analysis, due to space limitations, we design two sets of experiments, one for A, D → C and another for the symmetric A → D and D → A task, to examine the impact of the number of selected semantic neighbors, as illustrated in Fig. 6(a) and Fig. 6(b). We vary k_{sem} within set $\{0, 1, 2, 3, 4, 5\}$. By analyzing Fig. 6, we conclude that incorporating a moderate number of semantic neighbors effectively enhances the model's generalization ability in optimizing structure-semantic consistency, while excessive additions hinder it.

5.4 Ablation study

We design an ablation experiment to isolate the effectiveness of key components in TASSC. Specifically, we decouple the combination in Eq. (3) to shield the influence of both topological and semantic neighbors. Correspondingly, we reconstruct the regularization in Eq. (10) as $\mathcal{L}_{\text{tassc}}^{\text{no-sem}}$ and $\mathcal{L}_{\text{tassc}}^{\text{no-topo}}$, respectively.

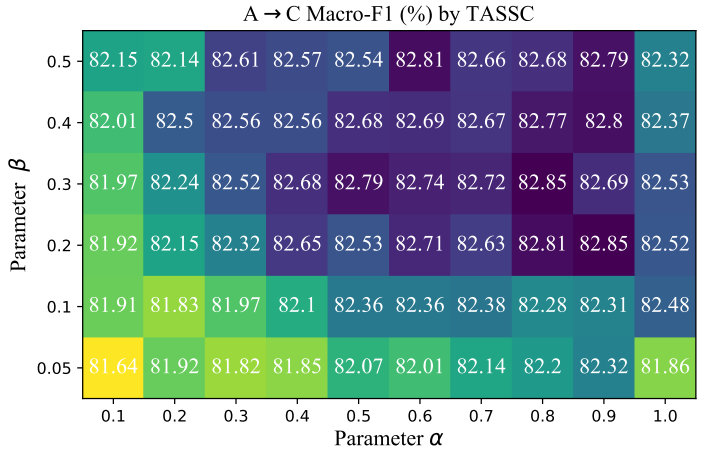
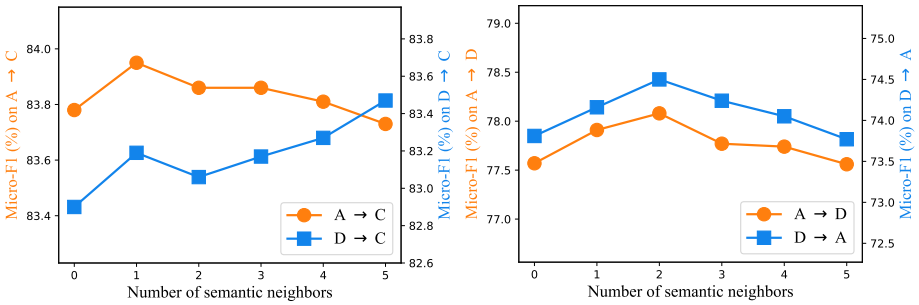


Fig. 5. The influence of parameters α and β on A → C task.



(a) Task A,D → C (b) Symmetric Task on A and D

Fig. 6. Micro-f1 performances (%) across different transfer tasks with varying numbers of semantic neighbors.

$\mathcal{L}_{\text{tassc}}^{\text{no-sem}}$ indicates that the positive samples constructed for v_t only utilize the target's local neighbors $\mathcal{N}_{\text{topo}}(v_t)$, excluding semantic neighbors. In contrast, $\mathcal{L}_{\text{tassc}}^{\text{no-topo}}$ is constructed by using semantic neighbors while excluding the topological neighbors. Here we use the classification loss $\mathcal{L}_{\text{cls}}$ and domain alignment loss $\mathcal{L}_{\text{align}}$ to regulate the training of TASSC and take it as the baseline. Then we use $\mathcal{L}_{\text{tassc}}^{\text{no-sem}}$, $\mathcal{L}_{\text{tassc}}^{\text{no-topo}}$ and $\mathcal{L}_{\text{tassc}}$ to form three variation methods.

As shown in Tab. 5, we observe that the variant methods incorporating target topology preservation or semantic constraint outperform the baseline. Furthermore, enforcing both target structure and semantic consistency, represented by $\mathcal{L}_{\text{tassc}}$, significantly improves performance. This phenomenon indicates that the designed losses have a positive impact on the final results.

Table 5. Ablation results (Avg. %) of TASSC on the three datasets. the bold means the best result and ✓marks the available regularization term.

$\mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{align}}$	$\mathcal{L}_{\text{tassc}}^{\text{no-sem}}$	$\mathcal{L}_{\text{tassc}}^{\text{no-topo}}$	$\mathcal{L}_{\text{tassc}}$	ArnetMiner		Airport		Twitch	
				Mi-F1	Ma-F1	Mi-F1	Ma-F1	Mi-F1	Ma-F1
✓				77.47	76.30	50.90	49.16	60.95	57.09
✓	✓			78.54	78.18	53.65	51.97	61.15	58.80
✓		✓		78.22	77.84	53.79	51.58	60.90	58.68
✓			✓	**78.84**	**78.53**	**55.28**	**53.72**	**61.56**	**59.67**

6 Conclusion

In this paper, we investigate the problem of UGDA by addressing the limitations of existing methods in exploiting the intrinsic interplay between structural proximity and semantic similarity within the target graph. To this end, we propose TASSC, a novel method that enforces consistency learning between structure and semantics in the feature space for enhancing domain generalization. Specifically, TASSC integrates topological and semantic neighborhood information to improve representation alignment and incorporate two self-supervised regularizations to enforce both global and local consistency. Furthermore, we introduce a scale-aware adaptive module that dynamically adjusts knowledge transfer based on domain scale discrepancies, enhancing target node representations for more effective consistency learning. Extensive experiments on three widely used datasets validate the effectiveness of TASSC, demonstrating significant performance compared to state-of-the-art baselines.

References

1. Bi, W., Cheng, X., Xu, B., Sun, X., Xu, L., Shen, H.: Bridged-gnn: Knowledge bridge learning for effective knowledge transfer. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. pp. 99–109 (2023)
2. Cai, R., Wu, F., Li, Z., Wei, P., Yi, L., Zhang, K.: Graph domain adaptation: A generative view. ACM Transactions on Knowledge Discovery from Data **18**(3), 1–24 (2024)
3. Chen, W., Ye, G., Wang, Y., Zhang, Z., Zhang, L., Wang, D., Zhang, Z., Zhuang, F.: Smoothness really matters: A simple yet effective approach for unsupervised graph domain adaptation. arXiv preprint arXiv:2412.11654 (2024)
4. Dai, Q., Wu, X.M., Xiao, J., Shen, X., Wang, D.: Graph transfer learning via adversarial domain adaptation with graph convolution. IEEE Transactions on Knowledge and Data Engineering **35**(5), 4908–4922 (2022)
5. Dantas, C.F., Gaetano, R., Ienco, D.: Semi-supervised heterogeneous domain adaptation via disentanglement and pseudo-labelling. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 440–456. Springer (2024)

6. Englert, B.B., Piva, F.J., Kerssies, T., De Geus, D., Dubbelman, G.: Exploring the benefits of vision foundation models for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1172–1180 (2024)
7. Gong, C., Li, X., Yu, J., Cheng, Y., Tan, J., Yu, C.: Self-pro: A self-prompt and tuning framework for graph neural networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 197–215. Springer (2024)
8. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. Advances in neural information processing systems **30** (2017)
9. Krause, A., Perona, P., Gomes, R.: Discriminative clustering by regularized information maximization. Advances in neural information processing systems **23** (2010)
10. Lei, P.I., Chen, X., Sheng, Y., Liu, Y., Guo, J., Gong, Z.: Gradual domain adaptation for graph learning. arXiv preprint arXiv:2501.17443 (2025)
11. Liu, M., Fang, Z., Zhang, Z., Gu, M., Zhou, S., Wang, X., Bu, J.: Rethinking propagation for unsupervised graph domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 13963–13971 (2024)
12. Liu, M., Zhang, Z., Ma, N., Gu, M., Wang, H., Zhou, S., Bu, J.: Structure enhanced prototypical alignment for unsupervised cross-domain node classification. Neural Networks **177**, 106396 (2024)
13. Liu, M., Zhang, Z., Tang, J., Bu, J., He, B., Zhou, S.: Revisiting, benchmarking and understanding unsupervised graph domain adaptation. Advances in Neural Information Processing Systems **37**, 89408–89436 (2025)
14. Liu, S., Li, T., Feng, Y., Tran, N., Zhao, H., Qiu, Q., Li, P.: Structural re-weighting improves graph domain adaptation. In: International conference on machine learning. pp. 21778–21793. PMLR (2023)
15. Liu, S., Zou, D., Zhao, H., Li, P.: Pairwise alignment improves graph domain adaptation. In: Proceedings of the 41st International Conference on Machine Learning. pp. 32552–32575 (2024)
16. Longa, A., Azzolin, S., Santin, G., Cencetti, G., Liò, P., Lepri, B., Passerini, A.: Explaining the explainers in graph neural networks: a comparative study. ACM Computing Surveys **57**(5), 1–37 (2025)
17. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
18. Pang, J., Wang, Z., Tang, J., Xiao, M., Yin, N.: Sa-gda: Spectral augmentation for graph domain adaptation. In: Proceedings of the 31st ACM international conference on multimedia. pp. 309–318 (2023)
19. Ribeiro, L.F., Saverese, P.H., Figueiredo, D.R.: struc2vec: Learning node representations from structural identity. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 385–394 (2017)
20. Rozemberczki, B., Allen, C., Sarkar, R.: Multi-scale attributed node embedding. Journal of Complex Networks **9**(2), cnab014 (2021)
21. Shen, X., Dai, Q., Chung, F.l., Lu, W., Choi, K.S.: Adversarial deep network embedding for cross-network node classification. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 2991–2999 (2020)
22. Shen, X., Pan, S., Choi, K.S., Zhou, X.: Domain-adaptive message passing graph neural network. Neural Networks **164**, 439–454 (2023)
23. Shi, B., Wang, Y., Guo, F., Xu, B., Shen, H., Cheng, X.: Graph domain adaptation: Challenges, progress and prospects. arXiv preprint arXiv:2402.00904 (2024)

24. Shui, C., Li, X., Qi, J., Jiang, G., Yu, Y.: Hierarchical graph contrastive learning for review-enhanced recommendation. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 423–440. Springer (2024)
25. Stubbemann, M., Stumme, G.: The mont blanc of twitter: Identifying hierarchies of outstanding peaks in social networks. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 177–192. Springer (2023)
26. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: Large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web. pp. 1067–1077 (2015)
27. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: Arnetminer: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 990–998 (2008)
28. Wang, H., Liu, G., Hu, P.: Tdan: Transferable domain adversarial network for link prediction in heterogeneous social networks. ACM Transactions on Knowledge Discovery from Data **18**(1), 1–22 (2023)
29. Wang, W., Zhang, G., Han, H., Zhang, C.: Correntropy-induced wasserstein gcn: Learning graph embedding via domain adaptation. IEEE Transactions on Image Processing **32**, 3980–3993 (2023)
30. Wu, J., He, J., Ainsworth, E.: Non-iid transfer learning on graphs. In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, pp. 10342–10350 (2023)
31. Wu, M., Pan, S., Zhou, C., Chang, X., Zhu, X.: Unsupervised domain adaptive graph convolutional networks. In: Proceedings of the web conference 2020. pp. 1457–1467 (2020)
32. Xiao, T., Zhu, H., Chen, Z., Wang, S.: Simple and asymmetric graph contrastive learning without augmentations. Advances in neural information processing systems **36**, 16129–16152 (2023)
33. Yin, N., Shen, L., Wang, M., Lan, L., Ma, Z., Chen, C., Hua, X.S., Luo, X.: Coco: A coupled contrastive framework for unsupervised domain adaptive graph classification. In: International Conference on Machine Learning. pp. 40040–40053. PMLR (2023)
34. You, Y., Chen, T., Wang, Z., Shen, Y.: Graph domain adaptation via theory-grounded spectral regularization. In: The Eleventh International Conference on Learning Representations (2023)
35. Zhang, P.: Evaluating accuracy of community detection using the relative normalized mutual information. Journal of Statistical Mechanics: Theory and Experiment **2015**(11), P11006 (2015)
36. Zhang, X., Du, Y., Xie, R., Wang, C.: Adversarial separation network for cross-network node classification. In: Proceedings of the 30th ACM international conference on information & knowledge management. pp. 2618–2626 (2021)
37. Zhong, Z., Mottin, D.: Efficiently predicting mutational effect on homologous proteins by evolution encoding. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 399–415. Springer (2024)