# SELECTIVITY MATTERS: SOURCE NODE INFLUENCE PRUNING FOR UNSUPERVISED GRAPH DOMAIN ADAPTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Unsupervised Graph Domain Adaptation (UGDA) aims to facilitate knowledge transfer from a labeled source graph to an unlabeled target graph by mitigating cross-domain distribution shifts. Existing methods primarily focus on node-level feature alignment in latent spaces, relying on the implicit assumption that all source nodes contribute positively to the alignment. However, this assumption often fails because a node's semantic information is intrinsically coupled with its topological graph structure. Due to structural shifts, source nodes with severe structural deviations (e.g., structural outliers) lack semantic counterparts in the target graph, and forcing alignment on them introduces severe noise and causes negative transfer. To bridge this gap, we argue that selective source node utilization is superior to full-graph training, thereby shifting the research paradigm from feature-level alignment to data-level refinement. To this end, we propose Source Node Influence Pruning (SNIP), a novel model-agnostic, data-centric refinement framework. Specifically, SNIP quantifies the structural discrepancy between individual source nodes and the target domain by integrating multiple centrality measures, assigning each source node an influence score. A rank-based normalization mechanism is further employed to eliminate scale variations across different measures, allowing SNIP to effectively identify and filter out structurally incompatible nodes with low influence scores. As a plug-and-play method, SNIP constructs a refined "sub-source" graph that is inherently more beneficial for subsequent alignment. Comprehensive experiments across eight transfer scenarios on five real-world datasets demonstrate that SNIP consistently outperforms competitive baselines and significantly enhances adaptation performance, validating the superiority of selective node utilization over full-graph training.

## 1 INTRODUCTION

The widespread adoption of Graph Neural Networks (GNNs) Kipf & Welling (2017); Yang et al. (2023) in social network analysis, bioinformatics, and recommender systems has highlighted the necessity of addressing distributional discrepancies in graph-structured data. These discrepancies, arising from varying data acquisition protocols or evolving topological structures, severely degrade model generalization. Unsupervised Graph Domain Adaptation (UGDA) Shi et al. (2025); Wilson & Cook (2020) aims to mitigate these distributional shifts, facilitating knowledge transfer from a labeled source graph to an unlabeled target graph.

Existing UGDA methods primarily focus on *feature-level alignment* in latent spaces, typically employing regularization techniques such as Maximum Mean Discrepancy (MMD) Shen et al. (2021), graph subtree discrepancy Wu et al. (2023), or adversarial learning Xiao et al. (2023) to enforce cross-domain consistency. While effective at mitigating feature-level shifts, these methods fundamentally rely on the implicit assumption that *all source nodes contribute positively to cross-domain knowledge transfer*, which is often invalid in practice. By treating the source graph as a whole, these methods attempt to force a global alignment with the target domain. However, in graph-structured data, a node's semantic information is intrinsically coupled with its topological structure. Due to pronounced structural shifts, source nodes exhibiting extreme topological properties, such as highly central hubs or structural outliers, often lack corresponding semantic counterparts in the target do-



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.

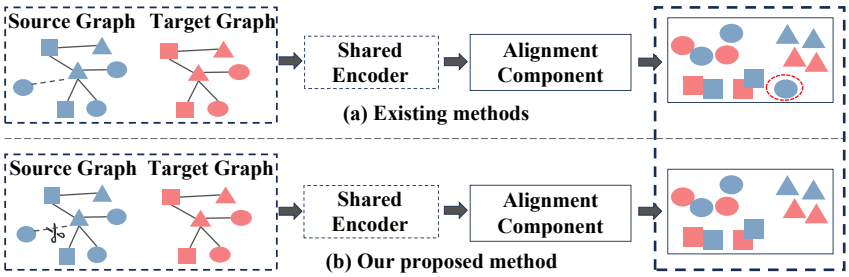



Figure 1: Conceptual comparison between existing feature-level alignment methods and our proposed pruning approach. The red dashed ellipse highlights how structural outliers distort the shared feature space and cause negative transfer, which is mitigated by our selective source node utilization.

main. Consequently, forcing the alignment on these structurally incompatible nodes inevitably injects significant noise, thereby distorting cross-domain representations and exacerbating negative transfer. As illustrated in Fig. 1, existing feature-level alignment methods typically utilize the entire source graph for alignment, making them highly vulnerable to the structurally incompatible nodes. For instance, outlier nodes in the feature space (circled in red dashed ellipse) can easily disturb the alignment process and cause the shared encoder to learn distorted representations for the target domain, degrading adaptation performance. In contrast, we argue that a "clean" source domain is a fundamental prerequisite for effective adaptation. In other words, **selective source node utilization is superior to full-graph training**. Guided by this philosophy, our goal is to identify and prune source nodes that are topologically incompatible with the target structure, thereby establishing a robust foundation for subsequent feature-based alignment.

To this end, we shift the research paradigm from conventional *feature-level alignment* to *data-level refinement*, moving from the question of "*how to align*" to "*what to align*". In this paper, we propose a novel data-centric refinement strategy named **S**ource **N**ode **I**nfluence **P**runing (SNIP). Instead of indiscriminately utilizing all source nodes, SNIP actively filters out structural noise by leveraging the target domain's global topological statistics as a reference benchmark. To achieve this, SNIP first quantifies the structural discrepancy between individual source nodes and the target domain to compute node influence scores, by integrating multi-dimensional centrality measures such as degree centrality and betweenness centrality. To mitigate the intrinsic scale variations among multiple centrality measures, a rank-based normalization mechanism is introduced, which converts the absolute values of each centrality measure into their corresponding relative percentile ranks within the source graph, thereby mapping diverse centrality measures into a uniform, non-parametric ordinal scale. Subsequently, SNIP ranks all source nodes based on their calibrated influence scores (i.e., the scores processed via the rank-based normalization mechanism), and filters out structurally incompatible source nodes with low scores. This process effectively eliminates source nodes that diverge significantly from the target domain's profile, constructing a refined "sub-source" graph that is inherently more beneficial to cross-domain alignment.

Since SNIP operates only at the data level and remains entirely model-agnostic, it can serve as a seamless, plug-and-play module for any existing UGDA method. We conduct extensive experiments using multiple backbone models on five real-world datasets, comprising three citation networks (i.e., ACMv9, Citationv1, and DBLPv7) and two Twitch gamer networks (i.e., Germany and England). Results demonstrate that SNIP consistently outperforms competitive baselines, significantly enhancing adaptation performance. Our contributions can be summarized as follows:

- **Data-Centric Refinement**: We challenge the conventional "align-all" paradigm, and propose an active source domain refinement framework. By identifying and pruning structural outliers, as a primary source of negative knowledge transfer, we shift the focus from "how to align" to "what to align".
- **Plug-and-Play Method**: We introduce a lightweight pruning strategy named SNIP, which can be seamlessly integrated into existing baselines to enhance their performance with minimal computational overhead.
- **Empirical Validation**: Extensive experiments on five public datasets demonstrate that pruning a calculated proportion of structurally incompatible source nodes yields significant improvements, thereby confirming the necessity of selective source node utilization.

## 2 Related Work

**Unsupervised Graph Domain Adaptation.** With the wide application of graph neural networks, Unsupervised Graph Domain Adaptation (UGDA) has become a key technology for addressing cross-domain distribution shifts in graph-structured data. Early studies primarily focused on learning domain-invariant node representations through adversarial alignment, such as UDAGCN Wu et al. (2020), ACDNE Shen et al. (2020), and ASN Zhang et al. (2021). Subsequent studies increasingly emphasized the role of graph structure in domain adaptation. JHGDA Shi et al. (2023) modeled graph hierarchy, while GRADE Wu et al. (2023) exploited subtree-level structural discrepancies. More recent studies have further explored how graph propagation, structural properties, and target-domain representations affect cross-domain generalization, including A2GNN Liu et al. (2024a), PairAlign Liu et al. (2024b), TDSS Chen et al. (2025), TASSC Zou et al. (2025), and DAGNN Yang et al. (2026). Beyond model-centric approaches, GraphAlign Huang et al. (2024) explored a data-centric direction by synthesizing a completely new, compact, and transferable graph through joint feature and topology optimization.

Despite these advances, existing methods generally treat source nodes as equally transferable and therefore overlook the heterogeneous contributions of individual source nodes to cross-domain transfer. Our SNIP instead evaluates node-level structural compatibility and selectively removes source nodes that are structurally incompatible with the target domain. Unlike GraphAlign Huang et al. (2024), which synthesizes a new graph, SNIP preserves the original source data and refines it by retaining a structurally compatible induced subgraph, providing a lightweight, model-agnostic, and plug-and-play data refinement strategy.

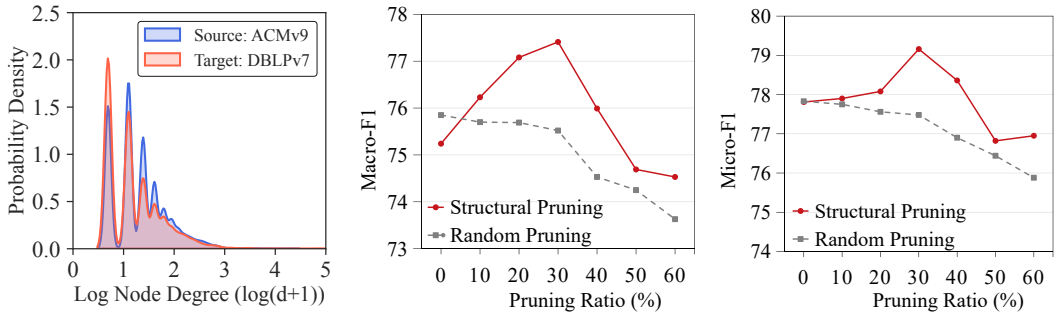
**Node Centrality**. Node centrality Park et al. (2026) is a key metric in network science used to quantify the importance of nodes in complex networks. Common measures, such as degree Bródka et al. (2011), betweenness Maurya et al. (2019), closeness Liu et al. (2023b), and eigenvector Frost (2023) centrality, describe a node's structural influence across multiple dimensions, ranging from local connectivity to global path dependence. While centrality measures are commonly used in GNNs for positional encoding or attention enhancement Li et al. (2024), we use them as structural discrepancy measures. By comparing source-node centralities with the target-domain structural profile, SNIP estimates node-level structural compatibility for cross-domain transfer.

**Graph Pruning**. Graph pruning Lin et al. (2026) has traditionally been used to reduce computational complexity and eliminate random noise. Advances in graph augmentation techniques have led to methods like DropEdge and graph sparsification Rong et al. (2020), which effectively mitigate over-smoothing and enhance model generalization. However, conventional pruning strategies are generally designed without considering cross-domain structural shifts and may therefore remove transferable nodes while retaining harmful ones. In contrast, SNIP explicitly incorporates target-domain structural information and selectively prunes source nodes that enlarge the domain discrepancy, making pruning directly serve the objective of domain adaptation.

## 3 Preliminaries

**Notations.** Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote an undirected, unweighted graph comprising $n = |\mathcal{V}|$ nodes and $m = |\mathcal{E}|$ edges, where $\mathcal{V}$ and $\mathcal{E}$ represent the node set and edge set, respectively. We present the graph data as a tuple $\mathcal{G} = (\mathbf{A}, \mathbf{X}, \mathbf{Y})$. Specifically, $\mathbf{A} \in \mathbb{R}^{n \times n}$ denotes the symmetric adjacency matrix, where $\mathbf{A}_{i,j} = 1$ if an edge $e_{i,j} \in \mathcal{E}$ exists between node $v_i$ and $v_j$, and $\mathbf{A}_{i,j} = 0$ otherwise. Furthermore, $\mathbf{X} = [x_1, x_2, \cdots, x_n]^\top \in \mathbb{R}^{n \times d}$ denotes the node attribute matrix, where $x_v \in \mathbb{R}^d$ represents the feature vector of node $v \in \mathcal{V}$, and $d$ represents the feature dimension. Analogously, $\mathbf{Y} = [y_1, y_2, \cdots, y_n]^\top \in \mathbb{R}^{n \times C}$ denotes the ground-truth label matrix, with $C$ representing the total number of categories.

**Problem Definition.** Unsupervised Graph Domain Adaptation (UGDA) facilitates knowledge transfer from a labeled source graph $\mathcal{G}^s = (\mathbf{A}^s, \mathbf{X}^s, \mathbf{Y}^s)$ to an unlabeled target graph $\mathcal{G}^t = (\mathbf{A}^t, \mathbf{X}^t)$. Under the generalized covariate shift assumption Ben-David et al. (2006; 2010), the two domains exhibit different feature distributions but share the same conditional label distribution: $\mathbb{P}_s(\mathbf{X}^s) \neq \mathbb{P}_t(\mathbf{X}^t)$ and $\mathbb{P}_s(\mathbf{Y}^s|\mathbf{X}^s) = \mathbb{P}_t(\mathbf{Y}^t|\mathbf{X}^t)$. The goal is to jointly exploit the labeled source graph $\mathcal{G}^s$ and unlabeled target graph $\mathcal{G}^t$ to learn a model that generalizes well to target-domain node classification.




(a) Distributional shifts. (b) Macro-F1 under various ratios. (c) Micro-F1 under various ratios.

Figure 2: Empirical motivation for selective source node utilization. (a) Kernel Density Estimation (KDE) curves across source (ACMv9) and target (DBLPv7) domains. (b–c) Performance comparison between "Structural Pruning" and "Random Pruning" under different pruning ratios.

# 4 METHODOLOGY

## 4.1 EMPIRICAL ANALYSIS

**Observation 1: Significant Cross-Domain Distributional Shifts.** We investigate cross-domain differences by visualizing the topological properties of the source (ACMv9) and target (DBLPv7) domains. Fig. 2a illustrates the Kernel Density Estimation (KDE) curves of log-transformed node degrees. It is observable that *the curves of ACMv9 and DBLPv7 exhibit significant deviations in peak positions and overall distribution shapes*, reflecting significant distributional shifts between domains. Even assuming perfect feature alignment, such structural mismatches persist. Enforcing alignment on nodes with deviant structures, such as outliers with extreme degrees, can induce negative transfer. Consequently, selective source node utilization is essential to retain only source nodes structurally compatible with the target domain.

**Observation 2: Negative Transfer Caused by Structurally Incompatible Nodes.** To validate the hypothesis that selective source node utilization is critical, we conduct a preliminary experiment utilizing Degree Centrality as the structural measure to prune a specified proportion of source nodes that exhibit significant discrepancies with the target domain, a strategy referred to as "Structural Pruning". Specific details and experimental settings are provided in Section 5.1. We compare "Structural Pruning" against "Random Pruning" under various pruning ratios across the source (ACMv9) and target (DBLPv7) domains, as illustrated in Fig. 2b and 2c. Random pruning exhibits performance decay as the pruning ratio increases, reflecting the indiscriminate removal of essential structural information. Conversely, *structural pruning demonstrates an initial performance improvement followed by a decline and yields improved Micro-F1 and Macro-F1 scores, reflecting the negative transfer phenomenon caused by structurally incompatible nodes*. Although pruning inevitably sacrifices some structural information, the above results indicate that the benefits of filtering structural noise outweigh the loss of raw topological structure. These findings support the premise that selective source node utilization is superior to full-graph training. Identifying and filtering out structurally incompatible source nodes significantly improves adaptation performance in the target domain.

## 4.2 SOURCE NODE INFLUENCE PRUNING FRAMEWORK

Building upon the above insights, we propose **S**ource **N**ode **I**nfluence **P**runing (**SNIP**), a lightweight, data-centric pruning strategy explicitly designed to filter out source nodes that are structurally incompatible with the target domain. By discarding these noisy nodes, SNIP prevents negative knowledge transfer effectively. As illustrated in Fig. 3, SNIP comprises three stages: (1) *Node Influence Measurement*, which quantifies the structural discrepancy between individual source nodes and the target domain to assign node influence scores, using multi-dimensional centrality measures; (2) *Rank-based Normalization*, which is a robust normalization mechanism to mitigate scale variation across multiple centrality measures; (3) *Structure Pruning*, which dynamically removes outlier nodes to mitigate negative knowledge transfer. The refined source graph is then utilized to train the GNN encoder, ensuring the model focuses on transferable structural patterns.

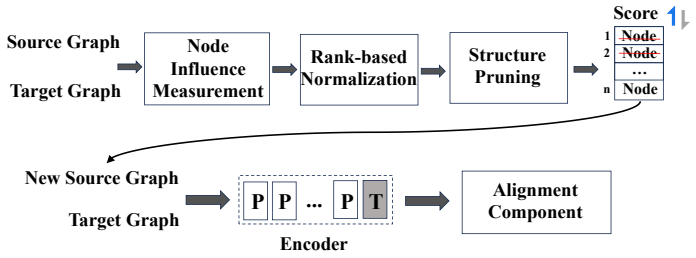



Figure 3: The overall pipeline of Source Node Influence Pruning (SNIP) framework.

**Node Influence Measurement.** To identify source nodes that are structurally incompatible with the target domain, we employ node influence scores as the pruning criterion. Specifically, we characterize node topological information via four complementary perspectives of centrality: *Degree*($d$), *Betweenness*($b$), *Closeness*($c$), and *Eigenvector*($e$), which capture diverse structural attributes ranging from local connectivity and routing bottlenecks to global accessibility and influential neighborhoods. Since these measures are foundational concepts in graph theory and complex network analysis, we omit their standard definitions here for brevity. Further details are provided in Appendix B.

Given source graph $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$ and target graph $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$, we first compute the four centrality metrics for all nodes of both graphs. Subsequently, we derive the global mean value for each centrality $i \in \{d, b, c, e\}$ within target domain $\mathcal{G}^t$. Let $\bar{C}_i^t$ denote this global mean, defined as: $\bar{C}_i^t = \frac{1}{|\mathcal{V}^t|} \sum_{v^t \in \mathcal{V}^t} C_i^t(v^t)$, where $C_i^t(v^t)$ denotes the value of centrality $i$ for target node $v^t$, and $|\mathcal{V}^t|$ is the number of target nodes. Then, we quantify the cross-domain structural discrepancy $\Delta C_i(v)$ for each source node $v \in \mathcal{V}^s$, defined as:

$$\Delta C_i^s(v) = |C_i^s(v) - \bar{C}_i^t|, \tag{1}$$

where $C_i^s(v)$ denotes the value of centrality $i$ for source node $v$, and "$|\cdot|$" represents taking the absolute value. Intuitively, the higher the value of structural discrepancy $\Delta C_i^s(v)$ for source node $v$, the lower its node influence. Source nodes with high $\Delta C_i^s$ values may be identified as structural outliers incompatible with the target domain's topology, which are more likely to induce negative transfer, as their neighborhood structures differ fundamentally from those in the target domain. Therefore, the node influence score of centrality $i$ for source node $v$ can be defined as:

$$S_i^s(v) = \frac{1}{\Delta C_i^s(v)}, \tag{2}$$

Although some global centrality measures (e.g., betweenness) can be computationally expensive, they are computed only once during an offline pre-processing step and introduce no training overhead. Furthermore, the measures are flexible, for large-scale graphs, they can be replaced with more efficient or linear-time measures to guarantee scalability.

**Rank-based Normalization.** Directly aggregating four raw node influence scores is problematic due to the vastly different scales of centrality metrics (e.g., degree is often an integer while betweenness is a small fraction). To ensure numerical robustness, we employ a Rank-based Normalization (RN) mechanism. For each centrality type $i$, all source nodes are first sorted by ascending order based on their node influence scores $S_i^s(v)$. Then, we can map these node influence scores into a normalized interval $[0, 1]$, defined as:

$$\hat{R}_i(v) = \frac{R_i(v) - 1}{|\mathcal{V}^s| - 1}, \tag{3}$$

where $R_i(v) \in \{1, \ldots, |\mathcal{V}^s|\}$ denotes the rank of node $v$ in terms of centrality $i$, and $\hat{R}_i(v)$ can serve as the new node influence score of centrality $i$ for source node $v$.

**Structure Pruning.** To comprehensively utilize the above four centrality metrics, the final node influence score is defined as the weighted sum of all normalized centrality ranks:

$$S(v) = \sum_{i \in \{d,b,c,e\}} w_i \cdot \hat{R}_i(v), \tag{4}$$

where $w_i$ is the weight coefficient for centrality $i$. A higher $S(v)$ indicates that node $v$ exhibits a topological pattern more similar to the target domain. All nodes in $\mathcal{V}^s$ are sorted in descending

order of $S(v)$ scores. According to a predefined pruning ratio $r$, the last $\lfloor r \cdot |\mathcal{V}^s| \rfloor$ nodes in the sorted sequence are removed from the source graph. This pruning process refines the source domain by retaining only those nodes that are structurally compatible with the target domain, thereby facilitating more effective domain adaptation. The overall process is shown in Algorithm 1 in Appendix A.

Since our SNIP operates only at the data level, it remains entirely model-agnostic, and can serve as a plug-and-play module for any existing baseline models, such as, AdaGCN Dai et al. (2023), A2GNN Liu et al. (2024a), etc. Taking the A2GNN backbone as an example, the overall objective based on Maximum Mean Discrepancy (MMD) Gretton et al. (2012) can be formulated as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \alpha \cdot \mathcal{L}_{\text{mmd}}, \tag{5}$$

where $\mathcal{L}_{\text{cls}}$ represents the cross-entropy loss for node classification on source graph, and $\mathcal{L}_{\text{mmd}}$ focuses on minimizing the distribution discrepancy between source and target features. The hyperparameter $\alpha$ balances the trade-off between classification and domain alignment.

### 4.3 Theoretical Analysis

We first revisit the graph domain adaptation bound adopted in prior studies Shen et al. (2018); You et al. (2023), then compare the complete target-risk upper bounds before and after node pruning and derive a necessary and sufficient condition under which SNIP strictly tightens the risk bound.

**Theorem 1.** Let $f$ and $c$ denote the feature extractor and classifier, and let $h = c \circ f \in \mathcal{H}$ be a GNN hypothesis, where $\mathcal{H}$ is the hypothesis space. Suppose that $f$ and $c$ are $K_f$- and $K_c$-Lipschitz continuous, where $K_f$ and $K_c$ are Lipschitz constants. Then, with probability at least $1-\delta$, the target domain risk $\epsilon_t(h, \hat{h})$ satisfies the following inequality Shen et al. (2018); You et al. (2023):

$$\epsilon_t(h, \hat{h}) \leq \widehat{\epsilon}_s(h, \hat{h}) + 2K_f K_c W_1\big(\mathbb{P}_s, \mathbb{P}_t\big) + \mathcal{R}\left(n_s, d, \delta\right) + \omega, \tag{6}$$

where $\widehat{\epsilon}_s(h, \hat{h})$ denotes the empirical source-domain risk, $\hat{h}$ is the labeling function, $W_1(\mathbb{P}_s, \mathbb{P}_t)$ is the 1-Wasserstein distance between source- and target domain distributions, and $\mathcal{R}(n_s, d, \delta) = \sqrt{\frac{4d}{n_s}\log\left(\frac{en_s}{d}\right) + \frac{1}{n_s}\log\left(\frac{1}{\delta}\right)}$ is the finite-sample complexity term. Here $n_s$ is the number of source nodes, $d$ is the pseudo-dimension of the hypothesis, and $\omega$ is the joint error of the ideal hypothesis.

Reducing the source-target discrepancy can tighten this bound. However, source-node pruning also reduces the effective source sample size and may simultaneously change the empirical source risk, the complexity term and the ideal joint error. Therefore, its benefit should be determined by comparing the complete bounds before and after pruning, where the subscripts "full" and "sub" denote the full-source and pruned sub-source settings, respectively. Let $\mathbb{P}_s$ and $\mathbb{P}'_s$ denote the source distributions before and after pruning, respectively, with $n'_s < n_s$. We define the reduction in the discrepancy-related term as the *alignment gain*:

$$\mathcal{T}_{\text{align}} = 2\Big[K_f^{\text{full}} K_c^{\text{full}} W_1(\mathbb{P}_s, \mathbb{P}_t) - K_f^{\text{sub}} K_c^{\text{sub}} W_1(\mathbb{P}'_s, \mathbb{P}_t)\Big]. \tag{7}$$

The remaining changes are defined as the *pruning cost* term:

$$\mathcal{T}_{\text{prune}} = \widehat{\epsilon}'_s(h_{\text{sub}}, \hat{h}) - \widehat{\epsilon}_s(h_{\text{full}}, \hat{h}) + \mathcal{R}(n'_s, d, \delta) - \mathcal{R}(n_s, d, \delta) + \omega_{\text{sub}} - \omega_{\text{full}}. \tag{8}$$

**Proposition 1.** Our SNIP strictly tightens the target-risk upper bound if and only if

$$\mathcal{T}_{\text{align}} > \mathcal{T}_{\text{prune}}. \tag{9}$$

Proposition 1 reveals the fundamental trade-off of source-node pruning: the reduction in source-target discrepancy must outweigh the cost caused by removing source nodes. SNIP explicitly targets this trade-off by assigning lower influence scores to source nodes with larger cross-domain structural discrepancies and preferentially pruning them. Therefore, under a predefined pruning ratio, SNIP is designed to increase the attainable alignment gain while avoiding indiscriminate information loss. The complete analysis and proof are provided in Appendix C.

This result also explains the importance of an appropriate pruning ratio. Insufficient source-node pruning may retain structurally incompatible source nodes, whereas excessive source-node pruning may increase the finite-sample penalty and discard informative source nodes. This theoretical observation is consistent with the empirical results in Fig. 2b, Fig. 2c and Fig. 4.

# 5 EXPERIMENTS AND RESULTS

## 5.1 EXPERIMENTAL SETTINGS

**Datasets.** We evaluate SNIP on five real-world datasets: three citation networks from Arnet-Miner Tang et al. (2008)[1], namely ACMv9 (A), Citationv1 (C), and DBLPv7 (D), and two Twitch gamer social networks Rozemberczki et al. (2021)[2], namely Germany (DE) and England (EN). Detailed dataset statistics are provided in Table 4 in Appendix D. We construct eight cross-domain adaptation scenarios: A→C, A→D, C→A, C→D, D→A, D→C, DE→EN and EN→DE.

**Baselines.** We compare SNIP with three groups of baselines: (1) Hypothesis transfer methods based on unsupervised graph representation learning, including DeepWalk Perozzi et al. (2014), node2vec Grover & Leskovec (2016), and ANRL Zhang et al. (2018); (2) Source-only GNN-based methods, including GAT Velickovic et al. (2018), GSAGE Hamilton et al. (2017), SGC Wu et al. (2019), GIN Xu et al. (2019) and GCN Kipf & Welling (2017); and (3) Graph domain adaptation methods, including CDNE Shen et al. (2021), AdaGCN Dai et al. (2023), ACDNE Shen et al. (2020), UDAGCN Wu et al. (2020), ASN Zhang et al. (2021), GRADE Wu et al. (2023), StruRW Liu et al. (2023a), SpecReg You et al. (2023), PairAlign Liu et al. (2024b), A2GNN Liu et al. (2024a), TDSS Chen et al. (2025), TASSC Zou et al. (2025), and DAGNN Yang et al. (2026).

**Implementation Details.** We train SNIP using the pruned source graph and the full target graph, and evaluate performance exclusively on target nodes. SNIP is implemented in PyTorch Paszke et al. (2019) with Adam optimization for up to 250 epochs and early stopping. We adopt AdaGCN Dai et al. (2023) and A2GNN Liu et al. (2024a) as backbone models using their official standard configurations, and A2GNN leverages a Gaussian kernel to compute the Maximum Mean Discrepancy (MMD) Gretton et al. (2012). We tune the pruning ratio $r \in \{10\%, 15\%, 20\%, 25\%\}$ and centrality trade-off coefficients $\{w_d, w_b, w_c, w_e\} \in [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$ via grid search. All experiments are conducted on a single NVIDIA GeForce RTX 3090 GPU, and Micro-F1 and Macro-F1 scores are averaged over three runs with different random seeds.

## 5.2 MAIN RESULTS

We report results on citation and social networks in Tables 1 and 2, with baseline results taken from Liu et al. Liu et al. (2024a) and Chen et al. Chen et al. (2025). The key observations are as follows:

Table 1: Overall performance (Micro-F1 / Macro-F1) on citation networks.

| Methods | A→C | | A→D | | C→A | | C→D | | D→A | | D→C | | Avg. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro |
| DeepWalk | 21.05 | 17.72 | 25.94 | 19.87 | 21.94 | 19.33 | 22.57 | 17.51 | 26.23 | 19.83 | 29.46 | 22.76 | 24.53 | 19.56 |
| node2vec | 29.89 | 25.84 | 24.54 | 19.50 | 21.76 | 17.99 | 28.95 | 24.98 | 28.61 | 22.05 | 21.16 | 16.22 | 25.82 | 21.08 |
| ANRL | 30.31 | 20.93 | 29.54 | 23.33 | 31.84 | 22.04 | 25.90 | 22.71 | 29.56 | 19.12 | 25.99 | 18.25 | 28.86 | 20.99 |
| GAT | 57.13 | 43.64 | 53.80 | 41.36 | 50.37 | 42.14 | 55.85 | 45.25 | 52.93 | 43.95 | 55.52 | 50.04 | 54.27 | 44.36 |
| GSAGE | 71.40 | 69.14 | 64.82 | 61.80 | 65.22 | 64.69 | 69.96 | 66.86 | 59.22 | 57.31 | 67.85 | 64.90 | 66.41 | 63.79 |
| SGC | 77.40 | 72.13 | 69.13 | 62.54 | 72.31 | 62.92 | 73.80 | 66.93 | 63.32 | 53.94 | 72.31 | 62.92 | 71.38 | 62.90 |
| GCN | 77.38 | 74.78 | 69.05 | 65.29 | 70.58 | 70.39 | 74.17 | 71.37 | 63.35 | 59.42 | 74.17 | 69.79 | 71.45 | 68.24 |
| UDAGCN | 72.15 | 60.33 | 66.95 | 64.83 | 66.80 | 67.22 | 71.77 | 69.46 | 58.16 | 55.89 | 73.28 | 61.12 | 68.18 | 63.14 |
| GRADE | 76.04 | 72.52 | 68.22 | 63.03 | 69.55 | 69.34 | 73.95 | 70.02 | 63.72 | 59.35 | 74.32 | 69.32 | 70.96 | 67.26 |
| CDNE | 78.76 | 76.83 | 71.58 | 69.24 | 74.22 | 75.06 | 74.36 | 71.34 | 69.62 | 70.45 | 78.88 | 77.36 | 74.57 | 73.38 |
| StruRW | 77.35 | 72.07 | 69.10 | 62.51 | 67.81 | 59.77 | 73.81 | 66.89 | 63.27 | 53.82 | 72.41 | 62.94 | 70.16 | 62.34 |
| PairAlign | 70.88 | 67.88 | 65.91 | 62.35 | 65.85 | 65.09 | 71.04 | 67.56 | 59.34 | 58.77 | 67.07 | 64.61 | 66.02 | 63.38 |
| SpecReg | 80.55 | 78.83 | 75.93 | 73.98 | 72.04 | 73.15 | 75.74 | 73.64 | 71.01 | 72.34 | 79.04 | 77.78 | 75.71 | 74.95 |
| ACDNE | 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 | 75.94 |
| TDSS | 83.20 | 81.40 | 79.54 | 78.13 | 76.94 | 78.33 | 78.96 | 77.55 | 74.89 | 76.46 | 82.29 | 80.75 | 79.54 | 79.06 |
| TASSC | 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 |
| DAGNN | 83.41 | 81.65 | 79.06 | 76.87 | 76.32 | 78.15 | 78.93 | 77.49 | 74.81 | 76.24 | 82.24 | 80.17 | 79.13 | 78.43 |
| AdaGCN | 79.32 | 76.51 | 75.04 | 71.39 | 69.67 | 69.47 | 75.59 | 72.34 | 69.67 | 69.47 | 78.20 | 74.22 | 74.91 | 72.45 |
| **SNIP$_{\text{AdaGCN}}$** | **83.03** | **81.16** | **76.86** | **75.05** | **76.60** | **77.26** | **76.74** | **75.34** | **73.59** | **74.06** | **80.67** | **79.55** | **77.92** | **77.07** |
| Improv.(%) | +3.71 | +4.65 | +1.82 | +3.66 | +6.93 | +7.79 | +1.15 | +3.00 | +3.92 | +4.59 | +2.47 | +5.33 | +3.01 | +4.62 |
| A2GNN | 82.75 | 80.86 | 77.84 | 75.18 | 74.63 | 76.31 | 78.28 | 75.90 | 73.44 | 74.94 | 81.09 | 78.84 | 77.95 | 76.72 |
| **SNIP$_{\text{A2GNN}}$** | **83.13** | **82.09** | **79.10** | **78.32** | **77.05** | **78.35** | **79.27** | **78.05** | **75.11** | **76.71** | **82.00** | **80.18** | **79.27** | **78.95** |
| Improv.(%) | +0.38 | +1.23 | +1.26 | +3.14 | +2.42 | +2.04 | +0.99 | +2.15 | +1.67 | +1.77 | +0.91 | +1.34 | +1.32 | +2.23 |

**(1) Incorporating SNIP consistently enhances baseline performance across diverse backbones.** As shown in Tables 1 and 2, SNIP consistently improves both AdaGCN and A2GNN across cita-

[1]https://github.com/yuntaodu/ASN/tree/main/data

[2]https://snap.stanford.edu/data/twitch-social-networks.html

Table 2: Overall performance (Micro-F1 / Macro-F1) on social networks.

| Methods | DE→EN | | EN→DE | | Avg. | |
|---|---|---|---|---|---|---|
| | Micro | Macro | Micro | Macro | Micro | Macro |
| node2vec | 52.64 | 46.96 | 54.61 | 50.10 | 53.62 | 48.53 |
| DeepWalk | 52.18 | 46.54 | 55.08 | 49.97 | 53.63 | 48.25 |
| GAT | 54.84 | 49.50 | 43.65 | 40.08 | 49.24 | 44.79 |
| GCN | 54.77 | 54.55 | 62.03 | 51.04 | 58.40 | 52.79 |
| GIN | 52.39 | 49.91 | 55.26 | 44.26 | 53.82 | 47.08 |
| UDAGCN | 59.74 | 58.19 | 58.69 | 56.35 | 59.21 | 57.27 |
| ACDNE | 58.08 | 56.31 | 58.79 | 57.92 | 58.43 | 57.11 |
| ASN | 55.45 | 51.21 | 60.45 | 45.90 | 57.95 | 48.55 |
| GRADE | 56.40 | 56.38 | 61.18 | 56.83 | 58.79 | 56.60 |
| SpecReg | 56.43 | 50.30 | 61.45 | 46.13 | 58.94 | 48.21 |
| TASSC | 58.74 | 57.08 | 64.38 | 62.25 | 61.56 | 59.67 |
| DAGNN | 61.05 | 60.28 | 65.42 | 62.93 | 63.24 | 61.61 |
| AdaGCN | 56.03 | 54.57 | 59.99 | 58.87 | 58.01 | 56.72 |
| **SNIP**$_{\text{AdaGCN}}$ | **56.65** | **55.31** | **62.70** | **60.41** | **59.67** | **57.86** |
| Improv.(%) | +0.62 | +0.74 | +2.71 | +1.54 | +1.66 | +1.14 |
| A2GNN | 59.54 | 58.44 | 63.90 | 61.42 | 61.72 | 59.93 |
| **SNIP**$_{\text{A2GNN}}$ | **60.67** | **59.13** | **65.21** | **63.08** | **62.94** | **61.10** |
| Improv.(%) | +1.13 | +0.69 | +1.31 | +1.66 | +1.22 | +1.17 |

tion and social networks, demonstrating strong plug-and-play compatibility. On citation networks, SNIP$_{\text{A2GNN}}$ improves A2GNN by an average of 1.32% in Micro-F1 and 2.23% in Macro-F1, achieving average Micro-F1 and Macro-F1 scores of 79.27% and 78.95%, respectively. It also achieves the best or second-best performance in all 12 reported metric values. Similarly, SNIP$_{\text{AdaGCN}}$ improves AdaGCN by 3.01% in Micro-F1 and 4.62% in Macro-F1 on average, with the largest gains of 6.93% and 7.79% in the C→A scenario. Consistent improvements are also observed on the two social networks. These results indicate that selectively pruning structurally incompatible source nodes provides a robust refinement strategy across different backbones and graph domains.

**(2) Graph domain adaptation methods generally outperform traditional hypothesis transfer methods and source-only GNN-based methods.** Tables 1 and 2 show that traditional embedding techniques (e.g., DeepWalk and node2vec) and source-only GNNs (e.g., GCN, GAT, and GIN) generally underperform graph domain adaptation methods under cross-domain distribution shifts. For example, on citation networks, DeepWalk and node2vec achieve average Micro-F1 scores of only 24.53% and 25.82%, respectively; in contrast, recent UGDA methods such as TDSS achieve around 79% average Micro-F1. Similar trends are observed on social networks. This is expected because these methods do not explicitly account for discrepancies between source and target graphs, whereas UGDA methods are specifically designed to mitigate feature and structural shifts.

### 5.3 Ablation Study

We conduct an ablation study on SNIP$_{\text{A2GNN}}$ to assess the contribution of each core component. we remove one component at a time and report Micro-F1 and Macro-F1 across multiple transfer scenarios in Table 3. The evaluated variants are: (1) $\sim$ w/o. D: Degree Centrality is removed. (2) $\sim$ w/o. B: Betweenness Centrality is removed. (3) $\sim$ w/o. C: Closeness Centrality is removed. (4) $\sim$ w/o. E: Eigenvector Centrality is removed. (5) $\sim$ w/o. R: Rank-based Normalization is removed.

Table 3: Ablation study across six transfer scenarios within citation network datasets.

| Methods | A→C | | A→D | | C→A | | C→D | | D→A | | D→C | | Avg. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro |
| SNIP$_{\text{A2GNN}}$ | **83.13** | **82.09** | **79.10** | **78.32** | **77.05** | **78.35** | **79.27** | **78.05** | **75.11** | **76.71** | **82.00** | **80.18** | **79.27** | **78.95** |
| $\sim$ w/o. D | 82.89 | 81.99 | 77.57 | 75.92 | 76.70 | 78.23 | 79.05 | 77.33 | 72.80 | 74.07 | 81.96 | 80.11 | 78.50 | 77.94 |
| $\sim$ w/o. B | 82.94 | 81.58 | 78.28 | 76.05 | 76.37 | 77.89 | 79.21 | 77.98 | 74.96 | 76.63 | 82.04 | 80.53 | 78.97 | 78.44 |
| $\sim$ w/o. C | 82.70 | 81.69 | 77.77 | 75.02 | 76.40 | 77.99 | 78.98 | 77.43 | 74.55 | 76.30 | 81.78 | 80.04 | 78.70 | 78.07 |
| $\sim$ w/o. E | 81.30 | 79.66 | 78.45 | 76.78 | 75.40 | 76.74 | 77.70 | 76.17 | 74.66 | 76.27 | 81.82 | 79.76 | 78.22 | 77.56 |
| $\sim$ w/o. R | 80.82 | 79.08 | 77.44 | 75.46 | 74.73 | 75.85 | 73.80 | 66.93 | 73.97 | 75.00 | 81.43 | 79.47 | 77.03 | 75.29 |

As shown in Table 3, removing any component degrades the average performance of SNIP. Excluding Degree, Betweenness, Closeness, and Eigenvector centrality decreases Macro-F1 by 1.01%, 0.51%, 0.88%, and 1.39%, respectively, demonstrating their complementary contributions to structural characterization. Notably, removing Rank-based Normalization causes the largest degradation, reducing Micro-F1 and Macro-F1 by 2.24% and 3.66%, respectively. These results validate the effectiveness of all components and particularly highlight the importance of Rank-based Normalization for reliable node influence estimation.

### 5.4 Sensitivity Analysis and Visualization

This section analyzes the pruning-ratio sensitivity and visualizes the learned representations. Further analyses of the MMD weight $\alpha$ and alternative structural measures are provided in the Appendix E.

**Impact of Pruning Ratio** $r$. We analyze the sensitivity of $\text{SNIP}_{\text{A2GNN}}$ to the pruning ratio $r$, which controls the proportion of removed source nodes. We vary $r$ from 0. to 0.6 with a step size of 0.1, and report Micro-F1 and Macro-F1. As shown in Fig. 4, performance is relatively stable for small pruning ratios and reaches its best range around $r \in [0.1, 0.3]$. When $r > 0.3$, performance degrades noticeably, suggesting that excessive pruning removes informative structural information. Overall, moderate pruning provides an optimal trade-off between noise removal and structural preservation.

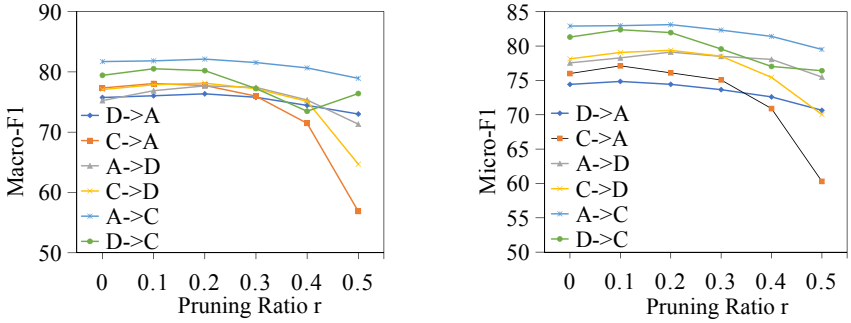
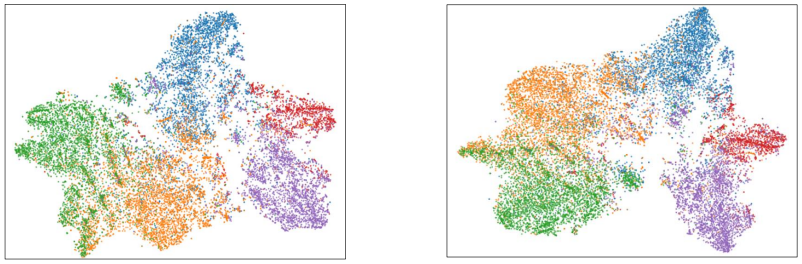



Figure 4: Sensitivity of Micro-F1 and Macro-F1 scores with different pruning rates $r$.

**Visualization.** To qualitatively examine the distribution alignment and feature discriminability between the source and target domains, we further employ t-SNE Hinton & Roweis (2002) to visualize the learned node representations under the C→A scenario. As shown in Fig. 5, the representations learned by $\text{SNIP}_{\text{A2GNN}}$ exhibit significantly clearer decision boundaries and superior cluster separation compared to the baseline A2GNN. Notably, SNIP achieves remarkable intra-class cohesiveness and sharper inter-class margins, effectively minimizing inter-cluster overlap. This visualization suggests that pruning structurally incompatible source nodes can reduce representation distortion and facilitate more effective cross-domain alignment.



(a) A2GNN (b) SNIP

Figure 5: T-SNE visualizations of the learned node representations under the C→A scenario.

## 6 Conclusion

This paper proposes Source Node Influence Pruning (SNIP), a data-centric refinement framework for Unsupervised Graph Domain Adaptation. SNIP is designed to identify and remove source nodes that are structurally incompatible with the target domain, thereby alleviating cross-domain structural distribution shifts and reducing their effects on knowledge transfer. Specifically, SNIP first calculates the influence scores of source nodes by integrating multiple centrality-based structural measures. It then ranks all source nodes in descending order according to their influence scores and removes a predefined proportion of nodes with the lowest scores. By selectively preserving source nodes that are more structurally compatible with the target domain, SNIP constructs a refined source graph for subsequent cross-domain alignment, improving adaptation performance. Notably, SNIP can be integrated into existing UGDA backbones in a plug-and-play manner, offering strong flexibility and scalability. Substantial experiments with different backbones across multiple transfer scenarios on five real-world datasets confirm that SNIP significantly improves graph domain adaptation performance, and provide a practical and general solution for cross-domain node classification.

### AI USE STATEMENT

Generative AI tools were used solely to improve the language, grammar, clarity, and readability of the manuscript. They were not used to develop research ideas, design the methodology or experiments, conduct data analysis, interpret experimental results, or develop theoretical analysis. All generative AI-assisted revisions were reviewed and verified by the authors, who take full responsibility for the final content of the manuscript.

### ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics and does not involve any apparent ethical risks.

### REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide the source code as supplementary materials. The experimental settings, including datasets, backbone models, evaluation metrics, hyperparameter configurations, and other implementation details, are described in the main text and Appendix. Detailed dataset statistics and additional experimental analyses are also provided in the Appendix. For the theoretical analysis, we state the proposition in the main text and provide complete derivations and proofs in the Appendix.

## REFERENCES

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, NIPS*, pp. 137–144, 2006. URL https://proceedings.neurips.cc/paper/2006/hash/b1b0432ceafb0ce714426e9114852ac7-Abstract.html.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2): 151–175, 2010. doi: 10.1007/S10994-009-5152-4. URL https://doi.org/10.1007/s10994-009-5152-4.

Piotr Bródka, Krzysztof Skibicki, Przemyslaw Kazienko, and Katarzyna Musial. A degree centrality in multi-layered social network. In *Proceedings of the International Conference on Computational Aspects of Social Networks, CASoN*, pp. 237–242, 2011. doi: 10.1109/CASON.2011.6085951. URL https://doi.org/10.1109/CASON.2011.6085951.

Wei Chen, Guo Ye, Yakun Wang, Zhao Zhang, Libang Zhang, Daixin Wang, Zhiqiang Zhang, and Fuzhen Zhuang. Smoothness really matters: A simple yet effective approach for unsupervised graph domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence, AAAI*, pp. 15875–15883, 2025. URL https://doi.org/10.1609/aaai.v39i15.33743.

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, 2023. URL https://doi.org/10.1109/TKDE.2022.3144250.

Hildreth Robert Frost. Eigenvector centrality for multilayer networks with dependent node importance. In *Proceedings of the Twelfth International Conference on Complex Networks and their Applications*, volume 1143, pp. 3–14, 2023. URL https://doi.org/10.1007/978-3-031-53472-0_1.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012. doi: 10.5555/2503308.2188410. URL https://dl.acm.org/doi/10.5555/2503308.2188410.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, 2016. URL https://doi.org/10.1145/2939672.2939754.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems*, pp. 1024–1034, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html.

Geoffrey E. Hinton and Sam T. Roweis. Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems 15, NIPS*, pp. 833–840, 2002. URL https://proceedings.neurips.cc/paper/2002/hash/6150ccc6069bea6b5716254057a194ef-Abstract.html.

Renhong Huang, Jiarong Xu, Xin Jiang, Ruichuan An, and Yang Yang. Can modifying data address graph domain adaptation? In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD*, pp. 1131–1142. ACM, 2024. doi: 10.1145/3637528.3672058. URL https://doi.org/10.1145/3637528.3672058.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the 5th International Conference on Learning Representations, ICLR*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Yangding Li, Shaobin Fu, Yangyang Zeng, Hao Feng, Ruoyao Peng, Jinghao Wang, and Shichao Zhang. Centrality-based relation aware heterogeneous graph neural network. *Knowledge-Based Systems*, 283:111174, 2024. URL https://doi.org/10.1016/j.knosys.2023.111174.

Fake Lin, Xi Zhu, Ziwei Zhao, Deqiang Huang, Yu Yu, Xueying Li, Zhi Zheng, Tong Xu, and Enhong Chen. Knowledge graph pruning for recommendation. *ACM Transactions on Information Systems*, 44(1):20:1–20:28, 2026. doi: 10.1145/3769107. URL https://doi.org/10.1145/3769107.

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 Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI*, pp. 13963–13971, 2024a. URL https://doi.org/10.1609/aaai.v38i12.29304.

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 2023 International Conference on Machine Learning, ICML*, volume 202, pp. 21778–21793, 2023a. URL https://proceedings.mlr.press/v202/liu23u.html.

Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In *Proceedings of the Forty-first International Conference on Machine Learning, ICML*, 2024b. URL https://openreview.net/forum?id=ttnbM598vZ.

Zhenfang Liu, Jianxiong Ye, and Zhaonian Zou. Closeness centrality on uncertain graphs. *ACM Transactions on the Web*, 17(4):29:1–29:29, 2023b. URL https://doi.org/10.1145/3604912.

Sunil Kumar Maurya, Xin Liu, and Tsuyoshi Murata. Fast approximations of betweenness centrality with graph neural networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM*, pp. 2149–2152, 2019. URL https://doi.org/10.1145/3357384.3358080.

Jin-Duk Park, Cong Tran, Won-Yong Shin, and Xin Cao. Centrality-based node feature augmentation for robust network alignment. *IEEE Transactions on Network Science and Engineering*, 13:1510–1526, 2026. URL https://doi.org/10.1109/TNSE.2025.3596908.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*, pp. 8024–8035, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*, pp. 701–710, 2014. URL https://doi.org/10.1145/2623330.2623732.

Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *Proceedings of the 8th International Conference on Learning Representations, ICLR*, 2020. URL https://openreview.net/forum?id=Hkx1qkrKPr.

Benedek Rozemberczki, Carl Allen, and Rik Sarkar. Multi-scale attributed node embedding. *Journal of Complex Networks*, 9(2), 2021. doi: 10.1093/COMNET/CNAB014. URL https://doi.org/10.1093/comnet/cnab014.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI*, pp. 4058–4065. AAAI Press, 2018. doi: 10.1609/AAAI.V32I1.11784. URL https://doi.org/10.1609/aaai.v32i1.11784.

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 Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI*, pp. 2991–2999, 2020. URL https://doi.org/10.1609/aaai.v34i03.5692.

Xiao Shen, Quanyu Dai, Sitong Mao, Fu-Lai Chung, and Kup-Sze Choi. Network together: Node classification via cross-network deep network embedding. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):1935–1948, 2021. URL https://doi.org/10.1109/TNNLS.2020.2995483.

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, CIKM*, pp. 2249–2258, 2023. URL https://doi.org/10.1145/3583780.3614928.

Boshen Shi, Yongqing Wang, Fangda Guo, Bing-Bing Xu, Hua-Wei Shen, and Xueqi Cheng. Domain adaptation for graph representation learning: Challenges, progress, and prospects. *Journal of Computer Science and Technology*, 40(2):283–300, 2025. URL https://doi.org/10.1007/s11390-024-4465-x.

Jie Tang, Jing Zhang, Limin Yao, Juanzi Li, Li Zhang, and Zhong Su. 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. URL https://doi.org/10.1145/1401890.1402008.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the 6th International Conference on Learning Representations, ICLR*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Garrett Wilson and Diane J. Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology*, 11(5):51:1–51:46, 2020. URL https://doi.org/10.1145/3400066.

Felix Wu, Amauri H. Souza Jr., Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML*, volume 97, pp. 6861–6871, 2019. URL http://proceedings.mlr.press/v97/wu19e.html.

Jun Wu, Jingrui He, and Elizabeth A. Ainsworth. Non-iid transfer learning on graphs. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI*, pp. 10342–10350, 2023. URL https://doi.org/10.1609/aaai.v37i9.26231.

Man Wu, Shirui Pan, Chuan Zhou, Xiaojun Chang, and Xingquan Zhu. Unsupervised domain adaptive graph convolutional networks. In *Proceedings of the web conference 2020, WWW*, pp. 1457–1467, 2020. URL https://doi.org/10.1145/3366423.3380219.

Zhiqing Xiao, Haobo Wang, Ying Jin, Lei Feng, Gang Chen, Fei Huang, and Junbo Zhao. SPA: A graph spectral alignment perspective for domain adaptation. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems*, pp. 37252–37272, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/754e80f98b2a141942f45a0eeb258a3c-Abstract-Conference.html.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Proceedings of the 7th International Conference on Learning Representations, ICLR*. OpenReview.net, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Chenxiao Yang, Qitian Wu, Jiahua Wang, and Junchi Yan. Graph neural networks are inherently good generalizers: Insights by bridging gnns and mlps. In *Proceedings of the Eleventh International Conference on Learning Representations, ICLR*, 2023. URL https://openreview.net/forum?id=dqnNW2omZL6.

Xiaoyu Yang, Junhai Zhai, and Jiankai Chen. DAGNN: disentangled asymmetric graph neural network for unsupervised graph domain adaptation. *Knowledge-Based Systems*, 346:116191, 2026. doi: 10.1016/J.KNOSYS.2026.116191. URL https://doi.org/10.1016/j.knosys.2026.116191.

Yuning You, Tianlong Chen, Zhangyang Wang, and Yang Shen. Graph domain adaptation via theory-grounded spectral regularization. In *Proceedings of the Eleventh International Conference on Learning Representations, ICLR*, 2023. URL https://openreview.net/forum?id=OysfLgrk8mk.

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 and Knowledge Management, CIKM*, pp. 2618–2626, 2021. URL https://doi.org/10.1145/3459637.3482228.

Zhen Zhang, Hongxia Yang, Jiajun Bu, Sheng Zhou, Pinggang Yu, Jianwei Zhang, Martin Ester, and Can Wang. ANRL: attributed network representation learning via deep neural networks. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI*, pp. 3155–3161, 2018. URL https://doi.org/10.24963/ijcai.2018/438.

Yan Zou, Yongzheng Lu, Na Li, Xiatian Zhu, Lan Du, Ming Yan, and Ying Ma. Target-adaptive structure-semantic consistency for unsupervised graph domain adaptation. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML-PKDD*, volume 16020 of *Lecture Notes in Computer Science*, pp. 182–198. Springer, 2025. doi: 10.1007/978-3-662-72243-5_11. URL https://doi.org/10.1007/978-3-662-72243-5_11.

## A ALGORITHM 1

**Algorithm 1** Source Node Influence Pruning

**Input**: Source graph $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$, Target graph $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$, weight coefficients $\{w_d, w_b, w_c, w_e\}$ for different centrality metrics, and pruning ratio $r$.
**Output**: Set of nodes to be removed $\mathcal{V}_{\text{prune}}$.

1: For source graph $\mathcal{G}^s$, compute node-level centrality scores: Degree $C_d^s(v)$, Betweenness $C_b^s(v)$, Closeness $C_c^s(v)$, and Eigenvector $C_e^s(v)$ for each node $v \in \mathcal{V}^s$.
2: For target graph $\mathcal{G}^t$, compute the global mean centrality values: $\bar{C}_d^t, \bar{C}_b^t, \bar{C}_c^t, \bar{C}_e^t$.
3: For each node $v \in \mathcal{V}^s$ and each metric $i \in \{d, b, c, e\}$, calculate the structural discrepancy from the target domain: $\Delta C_i^s(v) = |C_i^s(v) - \bar{C}_i^t|$, deriving the node influence score $S_i^s(v) = \frac{1}{\Delta C_i^s(v)}$.
4: **for** each metric $i \in \{d, b, c, e\}$ **do**
5: Generate ranks $R_i(v)$ for all $S_i^s(v)$ in ascending order.
6: Normalize the ranks to the range $[0, 1]$: $\hat{R}_i(v) = \frac{R_i(v)-1}{|\mathcal{V}^s|-1}$.
7: **end for**
8: For each node $v \in \mathcal{V}^s$, calculate the final node influence score $S(v)$ using the predefined weights: $S(v) = w_d \cdot \hat{R}_d(v) + w_b \cdot \hat{R}_b(v) + w_c \cdot \hat{R}_c(v) + w_e \cdot \hat{R}_e(v)$.
9: Sort all nodes $v \in \mathcal{V}^s$ in descending order based on $S(v)$.
10: Determine the number of source nodes to be pruned: $N_{\text{remove}} = \lfloor r \cdot |\mathcal{V}^s| \rfloor$.
11: Identify $\mathcal{V}_{\text{prune}}$ as the last $N_{\text{remove}}$ nodes in the sorted list (nodes with the lowest influence scores).
12: **return** $\mathcal{V}_{\text{prune}}$.

## B DEFINITIONS OF STRUCTURAL MEASURES.

Since our SNIP framework computes node influence scores based on multiple structural measures, we briefly introduce the candidate structural measures considered in this paper below:

- Degree Centrality indicates the direct connectivity of a node in the graph, measured by the number of edges directly connected to the node.
- Betweenness Centrality indicates the extent to which a node serves as a bridge or intermediary between other nodes, measured by the number or proportion of shortest paths between node pairs that pass through the node.
- Closeness Centrality indicates how efficiently a node can reach all other nodes in the graph, measured by the reciprocal of the average shortest-path distance from the node to all other reachable nodes.
- Eigenvector Centrality indicates the influence of a node by considering not only its connections but also the importance of its connected neighbors, measured by the corresponding value of the node in the principal eigenvector of the adjacency matrix.
- PageRank indicates the global importance of a node under a random-walk-based propagation process, measured by the probability that a random walker visits the node.
- Clustering Coefficient indicates the local clustering or cohesiveness around a node, measured by the ratio of existing edges among the node's neighbors to the total number of possible edges among those neighbors.
- $K$-Core Number indicates the structural coreness of a node within the graph, measured by the largest integer $K$ such that the node belongs to a $K$-core subgraph.
- Average Neighbor Degree indicates the average connectivity level and structural density of a node's local neighborhood, measured by the mean degree of all neighboring nodes that are directly connected to the node.

## C Complete Theoretical Analysis and Proofs

We first revisit the graph domain adaptation bound adopted in prior studies Shen et al. (2018); You et al. (2023). We then compare the complete target-risk upper bounds before and after pruning and derive a necessary and sufficient condition under which SNIP strictly tightens the bound.

**Theorem 1.** Let $f$ and $c$ denote the feature extractor and classifier, respectively, and let $h = c \circ f \in \mathcal{H}$ be a graph neural network hypothesis, where $\mathcal{H}$ is the hypothesis space of GNNs. Suppose that $f$ and $c$ are $K_f$- and $K_c$-Lipschitz continuous, where $K_f$ and $K_c$ are Lipschitz constants. Then, with probability at least $1-\delta$, the target domain risk $\epsilon_t(h, \hat{h})$ satisfies the following inequality Shen et al. (2018); You et al. (2023):

$$\epsilon_t(h, \hat{h}) \leq \widehat{\epsilon}_s(h, \hat{h}) + 2K_f K_c W_1\big(\mathbb{P}_s, \mathbb{P}_t\big) + \mathcal{R}\left(n_s, d, \delta\right) + \omega, \tag{10}$$

where $\widehat{\epsilon}_s(h, \hat{h})$ denotes the empirical source-domain risk, $\hat{h}$ is the labeling function, $W_1(\mathbb{P}_s, \mathbb{P}_t)$ is the 1-Wasserstein distance between source- and target domain distributions, and $\mathcal{R}(n_s, d, \delta) = \sqrt{\frac{4d}{n_s}\log\left(\frac{en_s}{d}\right) + \frac{1}{n_s}\log\left(\frac{1}{\delta}\right)}$ is the finite-sample complexity term. Here $n_s$ is the number of source nodes, $d$ is the pseudo-dimension of the hypothesis, and $\omega$ is the joint error of the ideal hypothesis.

Theorem 1 indicates that reducing the source-target discrepancy (i.e., $W_1(\mathbb{P}_s, \mathbb{P}_t)$) can tighten the target-risk upper bound. However, discrepancy reduction alone is insufficient to establish the theoretical benefit of our Source Node Influence Pruning. Pruning also reduces the effective source sample size and may change the empirical source risk, the complexity term and the ideal joint error. Therefore, the complete upper bounds before and after pruning must be compared, where the subscripts "full" and "sub" denote the full-source and pruned sub-source settings, respectively. By filtering out $k$ structural anomalies $\mathcal{V}_{\text{prune}} \subseteq \mathcal{V}^s$, SNIP yields a sub-source graph $\mathcal{G}^s_{\text{sub}} = \mathcal{G}^s \setminus \mathcal{V}_{\text{prune}}$ with $n'_s = n_s - k$ retained nodes. Let $\mathbb{P}'_s$ denote the corresponding distribution induced by $\mathcal{G}^s_{\text{sub}}$.

Applying Theorem 1 to the full source graph gives the upper bound

$$\mathcal{B}_{\text{full}} = \widehat{\epsilon}_s(h_{\text{full}}, \hat{h}) + 2K_f^{\text{full}} K_c^{\text{full}} W_1(\mathbb{P}_s, \mathbb{P}_t) + \mathcal{R}(n_s, d, \delta) + \omega_{\text{full}}, \tag{11}$$

whereas applying it to the retained sub-source graph gives

$$\mathcal{B}_{\text{sub}} = \widehat{\epsilon}'_s(h_{\text{sub}}, \hat{h}) + 2K_f^{\text{sub}} K_c^{\text{sub}} W_1(\mathbb{P}'_s, \mathbb{P}_t) + \mathcal{R}(n'_s, d, \delta) + \omega_{\text{sub}}. \tag{12}$$

We define the reduction in the discrepancy-related term as the *alignment gain*:

$$\mathcal{T}_{\text{align}} = 2\Big[K_f^{\text{full}} K_c^{\text{full}} W_1(\mathbb{P}_s, \mathbb{P}_t) - K_f^{\text{sub}} K_c^{\text{sub}} W_1(\mathbb{P}'_s, \mathbb{P}_t)\Big]. \tag{13}$$

The remaining changes are defined as the *pruning cost* term:

$$\mathcal{T}_{\text{prune}} = \widehat{\epsilon}'_s(h_{\text{sub}}, \hat{h}) - \widehat{\epsilon}_s(h_{\text{full}}, \hat{h}) + \mathcal{R}(n'_s, d, \delta) - \mathcal{R}(n_s, d, \delta) + \omega_{\text{sub}} - \omega_{\text{full}}. \tag{14}$$

The pruning cost accounts for the changes in empirical source risk, finite-sample complexity, and ideal joint error. In particular, because $n'_s < n_s$, source-node pruning generally increases the finite-sample complexity term $\mathcal{R}(\cdot, d, \delta)$.

**Proposition 1.** Our SNIP strictly tightens the target-risk upper bound if and only if

$$\mathcal{T}_{\text{align}} > \mathcal{T}_{\text{prune}}. \tag{15}$$

**Proof.** Subtracting Eq. equation 11 from Eq. equation 12 yields

$$\mathcal{B}_{\text{sub}} - \mathcal{B}_{\text{full}} = \mathcal{T}_{\text{prune}} - \mathcal{T}_{\text{align}}. \tag{16}$$

Therefore,

$$\begin{aligned} \mathcal{B}_{\text{sub}} < \mathcal{B}_{\text{full}} &\iff \mathcal{T}_{\text{prune}} - \mathcal{T}_{\text{align}} < 0 \\ &\iff \mathcal{T}_{\text{align}} > \mathcal{T}_{\text{prune}}. \end{aligned} \tag{17}$$

Hence, $\mathcal{T}_{\text{align}} > \mathcal{T}_{\text{prune}}$ is both a necessary and sufficient condition for our SNIP to strictly tighten the above target-risk upper bound. □

Proposition 1 shows that our pruning strategy does not unconditionally improve the generalization bound. Instead, the alignment gain induced by pruning must compensate for the finite-sample complexity penalty and the potential changes in the empirical source risk and ideal joint error.

We next relate this necessary and sufficient condition to the node-selection mechanism of SNIP. According to Eqs. 1 and 2, the influence score $S_i^s(v)$ of each source node $v$ is defined as a monotonically decreasing function of its cross-domain structural discrepancy $\Delta C_i(v)$. Consequently, source nodes that are structurally closer to the target domain receive higher influence scores, whereas source nodes with larger structural discrepancies receive lower scores. Our SNIP ranks the source nodes in descending order of their influence scores and prunes a predefined proportion of nodes from the end of the ranking. Since the influence score decreases monotonically with the structural discrepancy, this procedure is equivalent to preferentially pruning source nodes with the largest structural discrepancies across domains. Therefore, under a fixed pruning ratio, SNIP is designed to promote a positive discrepancy reduction and thereby increase the alignment gain $\mathcal{T}_{\text{align}}$ as much as possible.

For a simplified interpretation, suppose that the models before and after source-node pruning share the same Lipschitz constant upper bounds $\overline{K}_f$ and $\overline{K}_c$, and that source-node pruning introduces negligible changes in the empirical source risk and the ideal joint error. A sufficient condition to tighten the target-risk upper bound is

$$2\overline{K}_f\overline{K}_c\big[W_1(\mathbb{P}_s,\mathbb{P}_t) - W_1(\mathbb{P}'_s,\mathbb{P}_t)\big] > \mathcal{R}(n'_s,d,\delta) - \mathcal{R}(n_s,d,\delta). \tag{18}$$

This provides a direct interpretation of our SNIP strategy: source-node pruning is beneficial when the reduction in the source-target discrepancy outweighs the finite-sample penalty caused by the reduced source sample size. This result also explains why an appropriate pruning ratio is necessary. For detailed results, see Fig. 2b and 2c in Section 4.1, and Fig. 4 in Section 5.4. Insufficient pruning may retain structurally incompatible source nodes, whereas excessive pruning may discard informative nodes and introduce a pruning cost that exceeds the attainable alignment gain.

## D DATASETS

We evaluate our SNIP on five real-world datasets, including three citation networks from Arnet-Miner Tang et al. (2008): ACMv9 (A), Citationv1 (C), and DBLPv7 (D), alongside two social networks from the Twitch gamer networks Rozemberczki et al. (2021): Germany (DE) and England (EN). In the citation datasets, nodes represent academic papers and edges indicate citation relationships, whereas in the social networks, nodes and edges constitute users and their social connections, respectively. The detailed statistical properties of these datasets are summarized in Table 4.

Table 4: Detailed statistical properties of the citation and social network datasets.

| Types | Datasets | #Node | #Edge | #Feat | #Label |
|---|---|---|---|---|---|
| Citation Networks | ACMv9 | 9,360 | 15,556 | 6,775 | 5 |
| | Citationv1 | 8,935 | 15,098 | | |
| | DBLPv7 | 5,484 | 8,117 | | |
| Social Networks | Germany | 9,498 | 153,138 | 3,170 | 2 |
| | England | 7,126 | 35,324 | | |

## E FURTHER ANALYSIS

**Impact of Structural Measures**. In our SNIP framework, the quantification of node influence scores inherently depends on the combination of selected structural measures. To justify our selection rationale, we systematically compare two distinct configurations using A2GNN as the backbone. The default configuration combines degree centrality (D), betweenness centrality (B), closeness centrality (C), and eigenvector centrality (E). Conversely, the alternative configuration includes PageRank (PR), Clustering Coefficient (CC), K-Core Number (KC), and Average Neighbor Degree (AD), which primarily emphasize core-periphery structure, localized clustering or specific random-walk dynamics. Specifically, PageRank measures node importance through localized random-walk-based propagation; the Clustering Coefficient reflects how tightly neighbor nodes are connected;

Table 5: Comparison of different structural measure configurations across six transfer scenarios.

| Methods | A→C | | A→D | | C→A | | C→D | | D→A | | D→C | | Avg. | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro | Micro | Macro |
| A2GNN (Backbone) | 82.75 | 80.86 | 77.84 | 75.18 | 74.63 | 76.31 | 78.28 | 75.90 | 73.44 | 74.94 | 81.09 | 78.84 | 77.95 | 76.72 |
| Default (D, B, C, E) | 83.13 | **82.09** | **79.10** | **78.32** | **77.05** | **78.35** | **79.27** | **78.05** | 75.11 | 76.71 | **82.00** | **80.18** | **79.27** | **78.95** |
| Alternative (PR, CC, KC, AD) | **83.25** | 81.86 | 78.48 | 77.06 | 76.74 | 78.12 | 79.17 | 77.75 | **75.29** | **76.80** | 81.65 | 79.51 | 79.09 | 78.52 |

the K-core number characterizes a node's position in the graph's core–periphery hierarchy; and the Average Neighbor Degree measures the mean connectivity density of a node's immediate neighborhood. Further details are provided in Section 5.1.

As reported in Table 5, the empirical results across six cross-domain transfer scenarios validate the superiority of our default measure configuration. Specifically, the default configuration yields the highest average classification performance, delivering a Micro-F1 of 79.27% and a Macro-F1 of 78.95%, which outperforms the alternative configuration by margins of $0.18\%$ and $0.43\%$, respectively. Crucially, it is worth highlighting that the selection of structural measures within our SNIP framework is intrinsically decoupled and flexible, rather than rigidly fixed. The alternative configuration achieves superior scores in certain metrics (such as the Micro-F1 of A→C and D→A), indicating that measure substitution is feasible and demonstrating the high modularity of our SNIP approach. Since any custom set of graph structural measures can be seamlessly integrated without altering the core components, our SNIP framework exhibits excellent architectural extensibility. Consequently, when dealing with large-scale graphs where the calculation of global centrality metrics (e.g., betweenness) can be very time-consuming, researchers can flexibly deploy highly parallelizable or localized linear-time metrics (e.g., Average Neighbor Degree or PageRank approximations) as alternatives, which can still deliver competitive performance. This plug-and-play property successfully ensures the scalability of our approach.

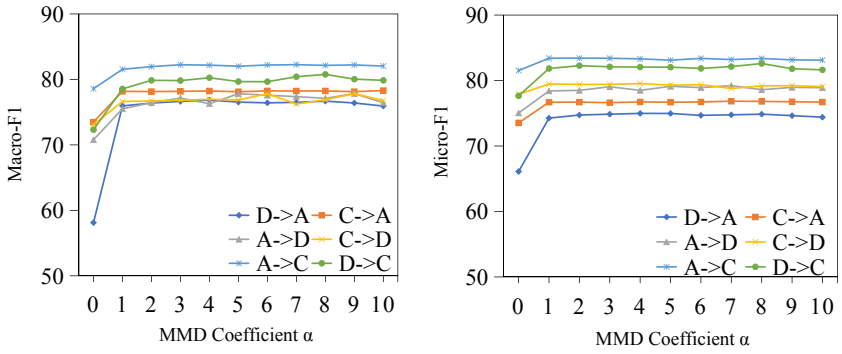



Figure 6: Sensitivity of Micro-F1 and Macro-F1 scores under different MMD weights $\alpha$ across six cross-domain adaptation scenarios on citation networks.

**Impact of trade-off coefficient $\alpha$ of MMD loss**. We investigate the sensitivity of the trade-off coefficient $\alpha$ across six distinct transfer scenarios, where $\alpha$ governs the regularization strength of cross-domain alignment. Specifically, we vary $\alpha$ over $[0, 1, 2, \ldots, 10]$ and evaluate its impact on SNIP with A2GNN as the backbone (i.e., $\text{SNIP}_{\text{A2GNN}}$). As illustrated in Fig. 6, the performance remains relatively stable over a broad range of $\alpha$, indicating that SNIP is not overly sensitive to the choice of this hyperparameter. We further consider the special case of $\alpha = 0$, in which the cross-domain alignment mechanism is completely removed. Under this setting, the model relies solely on the labeled source graph for supervision, and the overall objective reduces to the vanilla classification loss, i.e., $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}}$. A noticeable performance degradation is observed across all six transfer scenarios under this zero-alignment configuration. This consistent decline demonstrates the importance of explicitly reducing the source-target distribution discrepancy and suggests that source-node refinement alone is insufficient to fully address cross-domain shifts. Instead, our SNIP strategy provides a refined source graph that facilitates subsequent feature-level cross-domain alignment, and the two components jointly contribute to robust target-domain generalization.