Robust Cross Supervision With Target Mining for Source-Free Graph Domain Adaptation

Junyu Luo, Haoyu Tao, Xiao Luo, Yusheng Zhao, Zhiping Xiao, Dailan He, Wei Ju, Chong Chen, Xian-Sheng Hua, *Fellow, IEEE*, and Ming Zhang, *Member, IEEE*

***Abstract*—Graph domain adaptation has emerged as a critical challenge in real-world applications, where labeled graph data is often scarce and expensive to obtain. While existing methods have shown promise, they typically require access to source domain data, which may be restricted due to privacy concerns or data regulations. To address these limitations, we investigate the challenging yet practical problem of source-free graph domain adaptation. We propose a new approach named Robust Cross Supervision with Target Mining (ROSE) for this problem. ROSE achieves robustness by considering the complementary topology of graphs. The model consists of a message-passing branch for local semantic learning and a graph-kernel branch for global structural capture. Both branches are incorporated into a unified cross-supervision framework. To improve the robustness of the optimization process, we explore the context of the target domain, and divide the target data into discriminant set and anchor set. Then we incorporate the two tasks into a meta-learning optimization framework. Extensive experiments on benchmark datasets have demonstrated that our ROSE, compared with a wide range of baselines, always yields superior performance.**

***Index Terms*—Source-free domain adaptation, graph neural network, graph classification, distribution divergence.**

I. Introduction

GRAPH-STRUCTURED data has become increasingly prevalent in real-world applications, ranging from molecular generation [1] and object detection [2] to information retrieval [3]. Among various graph learning tasks, graph classification, which aims to categorize entire graph structures, has emerged as a fundamental challenge [4]. Graph Neural Networks (GNNs), built upon the message-passing paradigm, have demonstrated remarkable success in learning graph representations by iteratively aggregating neighborhood information and employing specialized pooling operations [5], [6]. However, when deployed in real-world scenarios, GNNs often encounter significant challenges, particularly in handling Out-of-Distribution (OOD) data and addressing label scarcity in target domains, as acquiring labeled graph data can be prohibitively expensive [7].

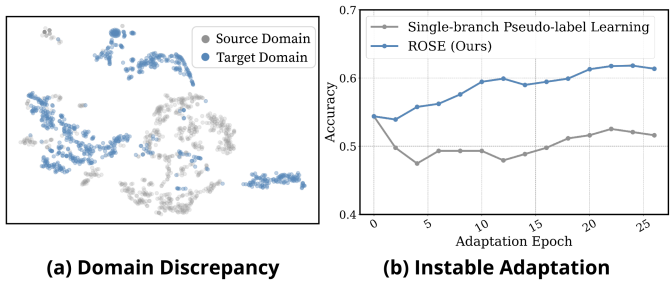


Fig. 1. Illustration of SFGDA challenges. **(a)** Visualization of significant structural and distributional discrepancy between source and target domains: the t-SNE projection of graph representations shows clear separation between domains with minimal overlap, indicating substantial domain shift; **(b)** Performance degradation and instability observed.

To address these challenges, Source-Free Domain Adaptation (SFDA) has emerged as an effective paradigm that tackles OOD problems without the access to source domain data [8], [9], [10], [11]. This approach is particularly valuable in scenarios where source data access is restricted due to privacy concerns, storage limitations, or regulatory constraints. While existing SFDA research has primarily focused on Euclidean data, particularly images [12], [13], its application to non-Euclidean graph data remains largely unexplored. The adaptation of SFDA to graph domains presents unique challenges due to the inherent complexity and high structural variance of graph data [14], [15], resulting in significant domain discrepancy, as illustrated in Fig. 1(a).

SFDA on graph data (SFGDA) faces two significant challenges due to the domain discrepancy. *(1) How to learn robust and discriminative graph representations capturing target-specific topology and features?* The scarcity of supervised signals, compounded by the complex interplay between node

Received 30 November 2024; revised 12 October 2025; accepted 16 January 2026. Date of publication 3 February 2026; date of current version 9 May 2026. The work of Ming Zhang was supported in part by the National Key Research and Development Program of China under Grant 2023YFC3341203 and in part by the National Natural Science Foundation of China under Grant 62276002. Recommended for acceptance by Zhen Wang. *(Corresponding authors: Xiao Luo; Zhiping Xiao; Ming Zhang.)*

Junyu Luo, Yusheng Zhao, Wei Ju, and Ming Zhang are with the State Key Laboratory for Multimedia Information Processing, School of Computer Science, Beijing Key Laboratory of Software and Hardware Cooperative Artificial Intelligence Systems, Peking University, Beijing 100871, China (e-mail: luojunyu@stu.pku.edu.cn; yusheng.zhao@stu.pku.edu.cn; juwei@pku.edu.cn; mzhang_cs@pku.edu.cn).

Haoyu Tao is with the School of Mathematical Sciences, Peking University, Beijing 100871, China (e-mail: tony_0501@stu.pku.edu.cn).

Xiao Luo is with the Department of Statistics, University of Wisconsin–Madison, Madison, WI 53706 USA (e-mail: xiao.luo@wisc.edu).

Zhiping Xiao is with the Department of Computer Science & Engineering, University of Washington, Seattle, WA 98195 USA (e-mail: patxiao@uw.edu).

Dailan He is with the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong (e-mail: hedailan@link.cuhk.edu.hk).

Chong Chen and Xian-Sheng Hua are with Terminus Group, Beijing 100010, China (e-mail: chenchong.cz@gmail.com; huaxiansheng@gmail.com).

The source code is available at https://github.com/luo-junyu/ROSE.

Digital Object Identifier 10.1109/TKDE.2026.3659905

features and graph structure inherent to non-Euclidean data, makes it difficult for the model to learn stable and effective representations without source guidance [16]. Additionally, GNNs are data-hungry, requiring substantial supervision signals [17]. Consequently, there is a critical need for methods that can effectively leverage limited target domain data while maximizing the extraction of structural and semantic information unique to graphs. *(2) How to ensure adaptation stability under significant domain shifts involving both graph structure and node attributes?*The large domain discrepancy, manifesting in both connectivity patterns and feature distributions, can lead to an unstable adaptation process. Errors may amplify through message passing across the graph structure, causing models to exhibit overconfidence in outlier samples or fail to maintain consistent performance. This necessitates a careful balance in the identification and utilization of different types of target samples: *discriminant samples* that help establish clear decision boundaries by reflecting target-specific structures, and *anchor samples* that maintain model stability throughout the adaptation process. The balance of these optimization tasks can contribute to the stabilization of the adaptation process. These interrelated challenges manifest in the poor and inconsistent performance of existing SFGDA methods, as evidenced by the empirical results shown in Fig. 1(b), highlighting the need for a more robust and comprehensive approach.

To address these challenges, we propose Robust Cross Supervision with Target Mining (ROSE), a new framework for SFDA on graphs. To learn robust and comprehensive graph representations in the target domain, ROSE extracts topology information from complementary perspectives through two specialized branches: a message-passing branch for capturing local semantic patterns, and a graph-kernel branch that employs random walk kernels and learnable hidden graphs to model global structural properties. These branches are unified through a cross-branch supervision mechanism that serves dual purposes: enforcing consistency regularization to prevent overfitting and facilitating information exchange between different representation spaces. To ensure adaptation stability, we explore the neighborhood of the unlabeled target samples and detect more *discriminant* graphs from the *anchor* graphs, using both purity and similarity metrics. At a high level, *discriminant* graphs serve to establish clear decision boundaries and extract distinctive features, while *anchor* graphs maintain model consistency throughout the adaptation process. These complementary roles are jointly optimized within a meta-learning framework, which dynamically balances their contributions to enhance adaptation stability. Extensive empirical evaluations demonstrate that ROSE consistently surpasses state-of-the-art methods across various benchmark datasets. The main contributions can be summarized as follows:

- We propose a comprehensive solution for SFGDA that addresses both representation learning and adaptation stability challenges through a unified framework.
- We develop a new architecture ROSE that combines message-passing and graph-kernel mechanisms, enhanced by cross-branch supervision and strategic sample role optimization through meta-learning.
- We conduct thorough empirical studies that demonstrate substantial improvements over existing methods, with detailed ablation studies validating our design choices and theoretical insights.

II. Related Work

Graph Neural Network: Graph Neural Networks (GNNs) have been effective tools in graph data analysis, demonstrating remarkable success in node classification, link prediction, and graph classification tasks [18], [19], [20], [21]. For graph classification, recent pooling methods capture richer distributional knowledge [22], [23]. Among various GNN architectures, Message-Passing Neural Networks (MPNNs) [24], [25] are dominant, iteratively updating node representations through neighborhood aggregation. However, MPNNs heavily rely on abundant labeled data and exhibit performance degradation when target domain labels are scarce [16]. Recent efforts enhance GNN expressivity and interpretability by incorporating subgraph-specific information [26] or learning global interactive patterns [27]. Complementary to MPNNs, graph kernel networks (GKNs) [28], [29] explicitly model graph topology, often leveraging subgraph structures, in an unsupervised manner. However, GKNs typically fall short of fully leveraging pre-trained supervision signals and neighborhood aggregation information. ROSE proposes a unified framework that synergistically combines the strengths of MPNNs and GKNs through cross-supervision, enabling effective extraction of target domain graph topology information for the challenging SFGDA scenario.

Graph Domain Adaptation: Graph domain adaptation (GDA) [7], [30] has emerged as a crucial research direction in machine learning, driven by the increasing need to transfer knowledge across different graph distributions in real-world applications [31], [32], [33]. While early GDA work addressed node-level tasks using discrepancy or adversarial methods [34], [35], recent efforts tackle graph-level adaptation. For instance, SLOGAN [7] utilizes causal discovery and generative intervention to disentangle robust causal features from spurious correlations for robust graph domain adaptation. A key limitation of conventional GDA is the requirement for source data access during adaptation, often infeasible due to privacy or storage constraints.

Source-free Graph Domain Adaptation: Source-free domain adaptation (SFDA), which assumes source data inaccessibility during adaptation, has gained significant attention due to its crucial role in data privacy protection [36], [37], [38], [39], [40]. Source-free graph domain adaptaion (SFGDA) [41] presents additional modeling challenges due to the inherent unique properties and structural complexity of graph data. Current SFGDA approaches can be broadly divided into two main streams: self-training approaches [42], [43] that leverage pseudo-labeling mechanisms, and generative approaches [44], [45], [46] that focus on distribution alignment. However, existing SFGDA methods exhibit critical limitations: (1) they struggle to extract comprehensive topology information from both local and global perspectives; (2) they fail to distinguish and leverage

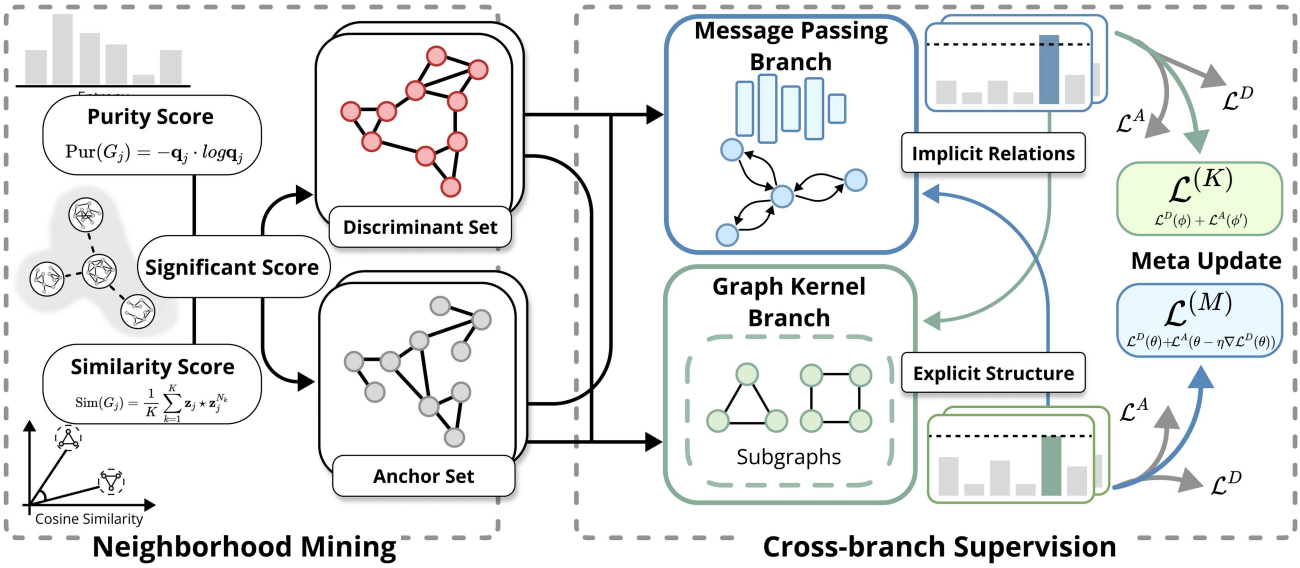


Fig. 2. Overview of the ROSE framework, depicting the components and dependencies within one iteration of the adaptation process. Neighborhood Mining (Section III-E), using significance scores derived from branch predictions, partitions target data into a Discriminant Set and an Anchor Set. These sets inform the two core representation learning branches: the Message-Passing Branch (Section III-C) for implicit local topology and the Graph-Kernel Branch (Section III-D) for explicit global structure. Cross-Branch Supervision and a Meta-Update mechanism (Section III-F) leverage these sets ($\mathcal{L}^D, \mathcal{L}^A$) to guide the optimization of both branches ($\mathcal{L}^{(K)}, \mathcal{L}^{(M)}$), facilitating robust and efficient source-free domain adaptation.

different types of target samples; and (3) they often lack proper regularization to prevent overfitting to noisy pseudo-labels. To address these challenges, ROSE leverages a dual-branch architecture, incorporating both message-passing and graph-kernel mechanisms to effectively capture and transfer the topological semantics of target graphs.

III. The Proposed ROSE

A. Problem Definition

Notations: Let $G = (V, E)$ denote a graph structure, where V denotes the node set and E denotes the edge set. The node attribute $v \in V$ is represented by a feature vector, collectively forming a node attribute matrix $\boldsymbol{X} \in \mathbb{R}^{|V| \times d^f}$, in which d^f represents the feature dimension.

Problem Setting: In the context of source-free domain adaptation for graphs, we consider:

- A labeled source domain $\mathcal{D}^{so} = \{(G_i^{so}, y_i^{so})\}_{i=1}^{N_s}$, where G_i^{so} denotes the i-th source graph and $y_i^{so} \in \{1, \ldots, C\}$ denotes its associated label;
- An unlabeled target domain $\mathcal{D} = \{G_j\}_{j=1}^{N}$, where G_j denotes the j-th target graph;
- A graph neural network Φ pre-trained on the source domain.

Key Challenges: While both domains have the identical label space $\{1, \ldots, C\}$, they exhibit different data distributions, *i.e.*, $P(G^{so}, y^{so}) \neq P(G, y)$. Crucially, during the adaptation process, the source graphs are completely inaccessible, and only the pre-trained model parameters are available for knowledge transfer. This setting introduces unique challenges as we need to adapt the model without access to the original training data while handling the complex topological structures inherent in graphs.

Objective: Our goal is to adapt the pre-trained model Φ to classify graphs in the target domain $\mathcal{D}$, utilizing only the model parameters and unlabeled target data, while addressing both the domain shift and the source-free constraints.

B. Framework Overview

Our paper investigates the problem of source-free graph domain adaptation (SFGDA) and proposes a new approach named ROSE for this problem, as illustrated in Fig. 2. Our methodology addresses the key SFGDA challenges outlined in Section I: learning robust target-specific representations (Challenge 1) and ensuring adaptation stability (Challenge 2). We first introduce two complementary branches for robust representation learning: the Message-Passing Branch (Section III-C) focusing on implicit local patterns, and the Graph-Kernel Branch (Section III-D) targeting explicit global structures, both crucial for learning robust target-specific representations. Subsequently, we detail the Neighborhood Mining strategy (Section III-E) which addresses the challenge of ensuring adaptation stability by identifying distinct roles for target samples (*discriminant* and *anchor* sets). Finally, we present the Robust Cross Supervision with Meta-Learning optimization framework (Section III-F) that integrates these components, using the mined sample sets to guide the collaborative training of the two branches for effective and stable adaptation, further tackling both representation learning and adaptation stability challenges.

C. Message-Passing Branch

Motivation: To initiate the process of learning robust graph representations (Challenge 1) without source data access, we first employ a message-passing branch that learns structural patterns implicitly through iterative neighborhood aggregation.

This design leverages the pre-trained source model and builds upon the success of message-passing neural networks in graph representation learning [5], [24], providing a foundation for adaptation.

Architecture: The message-passing branch consists of three key components: neighborhood aggregation, graph-level pooling, and classification. Specifically, for a node v with representation $\boldsymbol{h}_v^{(l)}$ at layer l, the representation updating follows:

$$\boldsymbol{h}_v^{(l)} = \mathrm{COM}^{(l)}\left(\boldsymbol{h}_v^{(l-1)}, \mathrm{AGG}^{(l)}\left(\left\{\boldsymbol{h}_u^{(l-1)}\right\}_{u\in\mathcal{N}(v)}\right)\right), \quad (1)$$

where $\mathrm{AGG}^{(l)}(\cdot)$ and $\mathrm{COM}^{(l)}(\cdot)$ denote the aggregation and combination operators, respectively, and $\mathcal{N}(v)$ represents the set of nodes adjacent to v.

After L layers of message passing, a readout function pools node-level features into a graph-level representation:

$$\boldsymbol{z} = \mathrm{READOUT}\left(\left\{\boldsymbol{h}_v^{(L)}\right\}_{v\in V}\right). \quad (2)$$

Finally, a classifier maps the graph representation to label distributions:

$$\boldsymbol{p}^{(M)} = \mathrm{CLA}(\boldsymbol{z}) = \Phi_\theta(G), \quad (3)$$

where θ denotes the learnable parameters of the message-passing branch $\Phi(\cdot)$. Here, topology implicitly guides the feature learning process via the aggregation mechanism. This branch is initialized with parameters pre-trained on the source domain, providing a foundation for target domain adaptation.

D. Graph-Kernel Branch

Motivation: While the message-passing branch provides a base representation, fully addressing the challenge of learning robust target-specific representations (Challenge 1) requires capturing target-specific global topology, especially under label scarcity. Message-passing networks face limitations here [16] and struggle with explicit substructures [47], [48], [49]. To overcome this, we introduce a complementary graph-kernel branch that explicitly models graph topology through learnable hidden graphs and random walk kernels [50], [51], enhancing the capture of diverse and potentially unique target domain structures.

Hidden Graph Construction: We introduce a set of M learnable hidden graphs $\{G'_m\}_{m=1}^M$ ($M = 10$ by default), where each hidden graph G'_m is parameterized by a learnable adjacency matrix and a node attribute matrix. These hidden graphs serve as structural templates to capture diverse topological patterns in the target domain.

Random Walk Kernel Computation: Given a graph pair $G = (V, E)$ and $G' = (V', E')$, we first construct their graph direct product $G_\times = (V_\times, E_\times)$, where:

$$V_\times = \{(v, v') : v \in V \wedge v' \in V'\},$$
$$E_\times = \{\{(v, v'), (u, u')\} : \{v, u\} \in E \wedge \{v', u'\} \in E'\}.$$

Following [50], [51], we compute the P-step random walk kernel incorporating node attributes:

$$k^{(P)}(G, G') = \sum_{p=0}^{P} \omega_p \mathbf{s}^\top \mathbf{A}_\times^p \mathbf{s}, \quad (4)$$

where $\mathbf{s} \in \mathbb{R}^{|V||V'|}$ is the flattened node attribute matrix $\boldsymbol{X}_\times = \boldsymbol{X}\boldsymbol{X}'^T$, ω_p are learnable coefficients, and $P = 3$ following [50].

Feature Generation and Classification: For each node v, we generate its r-hop subgraph G^v and compute kernel values with all hidden graphs:

$$\boldsymbol{f}_v = [\boldsymbol{f}_{v1}, \ldots, \boldsymbol{f}_{vM}], \text{ where } \boldsymbol{f}_{vm} = K(G^v, G'_m), \quad (5)$$

where K refers to the random walk kernel, which explicitly quantifies structural similarities.

The kernel features are then transformed and aggregated for classification:

$$\boldsymbol{b}_v = FFN(\boldsymbol{f}_v), \quad (6)$$

$$\Psi_\phi(G) = \mathrm{CLA}(\mathrm{READOUT}(\{\boldsymbol{b}_v\}_{v\in V})), \quad (7)$$

where ϕ denotes the learnable parameters in this branch.

Integration with Source-free Setting: The graph-kernel branch is designed to complement the message-passing branch without requiring source data access. It is initialized using pseudo-labels from the message-passing branch and collaboratively optimized during the adaptation process, providing explicit structural information that enhances the overall domain adaptation performance.

E. Neighborhood Mining for Discriminant Graphs

Motivation: To ensure adaptation stability (Challenge 2), differentiating the roles of target samples is crucial. Clearly, not all samples contribute equally [21], [52], [53]. Samples near decision boundaries are highly informative for refining the model but can introduce instability if overemphasized. These are termed *discriminant* samples. Conversely, samples confidently classified and located within class clusters provide stable learning signals, acting as *anchor* samples that prevent model drift. To effectively leverage both types, we propose a neighborhood-aware mining strategy to identify and distinctly utilize these discriminant and anchor samples, balancing rapid adaptation with stability.

Target Domain Mining: For each target sample G_j, we first generate its label distribution $\boldsymbol{p}_j$ from either branch. To construct a reliable confident set, we select samples with higher prediction confidence. Specifically, we define τ as the 40th percentile of confidence scores $\{\max_c \boldsymbol{p}_j[c]\}_{j=1}^N$ and construct the confident set as:

$$C = \{G_j | \max_c \boldsymbol{p}_j[c] > \tau\}, \quad (8)$$

where τ is adaptively determined to select the most confidently predicted samples. This equation establishes a dynamic filtering mechanism that adapts to the current model state during training.

Here, we assess the significance of each sample from two perspectives, *i.e.*, neighbor purity and similarity. Given a confident sample G_j, its K nearest neighborhood in the embedding

space is represented by $\{G_j^{N_1}, \ldots, G_j^{N_K}\}$. In this paper, K is empirically set to 10. Then, we calculate the proportion of samples for each class, *i.e.*,

$$\boldsymbol{q}_j[c] = \frac{1}{K}\sum_{k=1}^{K} \mathbf{1}\{\arg_{c'} \boldsymbol{p}_j^{N_k}[c'] = c\}, \tag{9}$$

where $\boldsymbol{p}_j^{N_k}$ denotes the label distribution of $G_j^{N_k}$. Then, the neighbor purity can be defined as the entropy of $\boldsymbol{q}_j$, *i.e.*,

$$\text{Pur}(G_j) = -\boldsymbol{q}_j \cdot \log \boldsymbol{q}_j. \tag{10}$$

These formulas quantify class diversity in a sample's neighborhood. High entropy ($\text{Pur}(G_j)$) indicates neighbors belong to multiple predicted classes, suggesting G_j is likely located near a decision boundary in the current feature space. Such samples are valuable because correctly classifying them requires the model to learn finer-grained distinctions.

Further, we calculate the average similarity between each graph sample and its neighborhood as:

$$\text{Sim}(G_j) = \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{z}_j \star \boldsymbol{z}_j^{N_k}, \tag{11}$$

where $\boldsymbol{z}_j$ and $\boldsymbol{z}_j^{N_k}$ denotes the deep features of G_j and $G_j^{N_k}$, respectively. $\star$ means the cosine similarity between two vectors.

Discriminant Set Construction: The key insight is that samples with high neighborhood purity (indicating decision boundary proximity) and high feature similarity (suggesting reliable neighborhood structure) are most valuable for adaptation. We combine these metrics to compute significance scores:

$$\text{Sig}(G_j) = \text{Pur}(G_j)\text{Sim}(G_j). \tag{12}$$

This multiplicative combination prioritizes samples that are both near decision boundaries (high purity/entropy, informationally rich for adaptation) and structurally coherent with their neighbors (high similarity, reducing noise). These form the discriminant set D.

Then C is partitioned into two subsets: a discriminant set D and an anchor set $A = C \setminus D$, where D is determined by the following process:

$$D = \{G_j | j \in \text{argsort}(-\text{Sig}(G_j))[1 : \lfloor \rho\% \cdot |C| \rfloor]\}, \tag{13}$$

where $\rho\%$ controls the proportion of discriminant samples (validated in Section IV-G). The remaining confident samples form the anchor set A, representing core, stable examples within predicted class clusters.

Integration with Adaptation: This strategy identifies samples crucial for adaptation. The discriminant set D, containing boundary samples, drives model refinement and rapid adaptation. The anchor set A, containing stable samples, provides consistent signals to prevent catastrophic forgetting and ensure stability. The meta-learning optimization (Section III-F) explicitly leverages these distinct roles to achieve both rapid and robust adaptation.

F. Robust Cross Supervision With Meta Learning

Motivation: To effectively utilize the mined samples and the dual-branch architecture for robust representation (Challenge 1) and stable adaptation (Challenge 2), we propose a novel optimization strategy. Cross-supervision between the diverse branches (implicit local vs. explicit global topology) mitigates confirmation bias inherent in traditional self-training [54], [55], [56], enhancing representation consistency. Crucially, to manage the stability-plasticity trade-off (Challenge 2), we employ meta-learning. This framework explicitly uses the anchor set to stabilize the learning process driven by the informative but potentially noisy discriminant set, preventing potential gradient conflicts and ensuring robust convergence.

Cross-Branch Supervision: Given predictions from the message-passing branch, we first identify confident samples $C^{(M)}$ and partition them into discriminant set $D^{(M)}$ and anchor set $A^{(M)}$ following Section III-E. These sets are then used to supervise the graph-kernel branch through two loss terms:

$$\mathcal{L}^D(\phi) = -\sum_{G_j \in D^{(M)}} \log\left(\boldsymbol{p}_j^{(K)}\left[\hat{y}_j^{(M)}\right]\right), \tag{14}$$

$$\mathcal{L}^A(\phi) = -\sum_{G_j \in A^{(M)}} \log\left(\boldsymbol{p}_j^{(K)}\left[\hat{y}_j^{(M)}\right]\right), \tag{15}$$

where $\boldsymbol{p}_j^{(K)}$ denotes predictions from the graph-kernel branch and $\hat{y}_j^{(M)}$ represents pseudo-labels from the message-passing branch.

Meta-Learning Optimization: A key aspect of ensuring adaptation stability (Challenge 2) is managing the potentially conflicting objectives of learning from discriminant samples (for rapid adaptation) and anchor samples (for stability). To address potential gradient conflicts ($\nabla\mathcal{L}^D(\phi) \cdot \nabla\mathcal{L}^A(\phi) < 0$) within the optimization of a single branch, we employ a meta-learning framework [57], [58]. We take the $\mathcal{L}^D(\phi)$ and $\mathcal{L}^A(\phi)$ as *meta-train* and *meta-test* tasks, respectively. This meta-learning structure dynamically adjusts the optimization focus between these two sub-tasks, aiming to prevent the pursuit of rapid adaptation via discriminant samples from destabilizing the model's performance on anchor samples, thereby achieving robust and rapid SFGDA. The inner loop is formulated as:

$$\phi' = \phi - \eta\nabla\mathcal{L}^D(\phi), \tag{16}$$

in which η denotes the learning rate. In the outer loop, we further optimize ϕ by minimizing the following equation:

$$\mathcal{L}^{(K)}(\phi) = \mathcal{L}^D(\phi) + \alpha\mathcal{L}^A(\phi'), \tag{17}$$

where α is used to balance loss, which is investigate in Section IV-G. In this way, we can promise that learning from $\mathcal{L}^A(\phi)$ would always benefit $\min_\phi \mathcal{L}^D(\phi)$ to provide a robust optimization procedure.

Then, the graph-kernel branch can provide extra supervision for the message-passing branch. Similarly, we have

$$\mathcal{L}^{(M)}(\theta) = \mathcal{L}^D(\theta) + \alpha\mathcal{L}^A(\theta - \eta\nabla\mathcal{L}^D(\theta)). \tag{18}$$

Discussion: Our method synergistically combines cross-supervision and meta-learning. Cross-supervision leverages diverse branch perspectives (implicit local vs. explicit global) to enhance representation robustness and reduce confirmation bias (addressing Challenge 1). Concurrently, the meta-learning framework specifically targets adaptation stability (Challenge 2). By treating the discriminant set loss $\mathcal{L}^D$ as the meta-train task and the anchor set loss $\mathcal{L}^A$ as the meta-test task (17), (18), the optimization ensures that updates driven by informative boundary samples ($\mathcal{L}^D$) do not degrade performance on stable core samples ($\mathcal{L}^A$). This explicitly uses the anchor set to enforce stability, managing potential conflicts and promoting robust convergence during adaptation.

Computational Complexity: ROSE is designed for efficient inference, leveraging only the message-passing branch with a standard GNN complexity of $\mathcal{O}(|\mathcal{A}|_0\, d + |\mathcal{V}|d^2)$. During the adaptation phase, the graph-kernel branch introduces an additional computational cost primarily associated with the kernel computations, approximately $\mathcal{O}(|\mathcal{V}|M)$, where M is the number of hidden graphs. This dual-branch structure maintains a practical balance between adaptation performance and computational demands, ensuring efficiency during inference. Empirical validation of time efficiency and scalability is presented in Section IV-E.

G. Theoretical Analysis

In this part, we demonstrate the robustness of our methods through theoretical analysis. We focus on two key aspects: (1) the effectiveness of discriminant set optimization and (2) the robustness advantage over traditional methods.

Risk Definitions: To formally analyze the properties of our meta-learning approach, we define several risk components based on the expected loss over the underlying data distributions. These definitions are crucial for establishing the theoretical guarantees presented in Theorems 3.1 and 3.2. Let $\mathcal{L}^{(M)}(\theta) = \mathcal{L}^D(\theta) + \alpha\mathcal{L}^A(\theta - \eta\nabla\mathcal{L}^D(\theta))$ be the overall meta-learning loss for one branch (e.g., the message-passing branch with parameters θ). We define the *true risk* $\mathcal{R}^{(M)}(\theta)$ as the expected loss over the true target data distribution p:

$$\mathcal{R}^{(M)}(\theta) = \mathbb{E}_{(\boldsymbol{X},y)\sim p}\mathcal{L}^{(M)}(\theta). \tag{19}$$

This represents the ideal performance objective we aim to minimize. Correspondingly, the *empirical risk* $\hat{\mathcal{R}}^{(M)}(\theta)$ is the expected loss over the observed pseudo-labeled target distribution p_γ:

$$\hat{\mathcal{R}}^{(M)}(\theta) = \mathbb{E}_{(\boldsymbol{X},\hat{y})\sim p_\gamma}\mathcal{L}^{(M)}(\theta). \tag{20}$$

This is the risk we can actually estimate and minimize during training, reflecting performance on potentially noisy pseudo-labels. The relationship between $\mathcal{R}^{(M)}(\theta)$ and $\hat{\mathcal{R}}^{(M)}(\theta)$, analyzed in Theorem 3.2, is key to understanding the robustness of our method to label noise. Let θ^* and $\hat{\theta}$ be the minimizer of the true risk $\mathcal{R}^{(M)}(\theta)$ and empirical risk $\hat{\mathcal{R}}^{(M)}(\theta)$ respectively.

Furthermore, to specifically analyze the effect of our neighborhood mining strategy, we define risks associated with the discriminant and anchor sets. Assuming the data in the discriminant set D and anchor set A follow distributions p_D and p_A respectively, we define the *discriminant risk* $\mathcal{R}^D(\theta)$ and *anchor risk* $\mathcal{R}^A(\theta)$:

$$\mathcal{R}^D(\theta) = \mathbb{E}_{(\boldsymbol{X},y)\sim p_D}\mathcal{L}^D(\theta), \tag{21}$$

$$\mathcal{R}^A(\theta) = \mathbb{E}_{(\boldsymbol{X},y)\sim p_A}\mathcal{L}^A(\theta), \tag{22}$$

and let θ_D^* and θ_A^* be the minimizer of the discriminant risk $\mathcal{R}^D(\theta)$ and anchor risk $\mathcal{R}^A(\theta)$ respectively. These definitions allow us to formally quantify the learning objectives associated specifically with the informative boundary samples ($\mathcal{R}^D(\theta)$) and the stable core samples ($\mathcal{R}^A(\theta)$). Theorem 3.1 utilizes these risks to demonstrate how our meta-learning framework effectively prioritizes learning from the discriminant set while being stabilized by the anchor set.

Effectiveness of Discriminant Set Optimization: Our first theorem demonstrates that our method achieves better optimization on the discriminant set:

Theorem 3.1 (Priority Learning): Under the following conditions:

$$\sum_{k=1}^{C}\mathbb{E}_{\boldsymbol{X}}(\mathcal{L}^{(M)}(\theta_D^*;\boldsymbol{X},k) - \mathcal{L}^{(M)}(\hat{\theta};\boldsymbol{X},k)) \le 0, \tag{23}$$

$$\mathbb{E}_{(\boldsymbol{X},y)\sim p}(\nabla\mathcal{L}^D(\hat{\theta})^T\nabla\mathcal{L}^A(\hat{\theta}) - \nabla\mathcal{L}^D(\theta_D^*)^T\nabla\mathcal{L}^A(\theta_D^*)) \le 0, \tag{24}$$

$$\mathcal{R}^A(\theta_D^*) + \mathcal{R}^A(\theta_A^*) \le 2\mathcal{R}^A(\hat{\theta}). \tag{25}$$

we have:

$$\mathcal{R}^D(\hat{\theta}) - \mathcal{R}^D(\theta_D^*) \le \alpha(\mathcal{R}^A(\hat{\theta}) - \mathcal{R}^A(\theta_A^*)) + \mathcal{O}(\eta^2). \tag{26}$$

Proof: Under the noisy label model with flip rate γ, the empirical risk can be expressed as $\hat{\mathcal{R}}^{(M)}(\theta) = \frac{C(1-\gamma)-1}{C-1}\mathcal{R}^{(M)}(\theta) + \frac{\gamma}{C-1}\mathbb{E}_{\boldsymbol{X}}\sum_{k=1}^{C}\mathcal{L}^{(M)}(\theta;\boldsymbol{X},k)$. Since $\hat{\mathcal{R}}^{(M)}(\hat{\theta}) \le \hat{\mathcal{R}}^{(M)}(\theta_D^*)$, we have $(C(1-\gamma)-1)(\mathcal{R}^{(M)}(\hat{\theta}) - \mathcal{R}^{(M)}(\theta_D^*)) \le \gamma\mathbb{E}_{\boldsymbol{X}}\sum_{k=1}^{C}(\mathcal{L}^{(M)}(\theta_D^*;\boldsymbol{X},k) - \mathcal{L}^{(M)}(\hat{\theta};\boldsymbol{X},k))$. Applying Taylor expansion to the meta-update term yields $\mathcal{R}^{(M)}(\theta) = \mathcal{R}^D(\theta) + \alpha\mathcal{R}^A(\theta) - \alpha\eta\mathbb{E}\nabla\mathcal{L}^D(\theta)^T\nabla\mathcal{L}^A(\theta) + \mathcal{O}(\eta^2)$. Combining these inequalities with the stated conditions completes the proof. □

This theorem shows that our method has a better optimization effect on discriminant set which are at the decision boundary and more informative. Therefore, this result confirms the effect of neighborhood mining in data utilization.

Robustness Analysis: Next, we demonstrate the robustness of our meta-learning method compared to the traditional approach. Let us introduce another loss function:

$$\mathcal{L}_0^{(M)}(\theta) = \mathcal{L}^D(\theta) + \alpha\mathcal{L}^A(\theta), \tag{27}$$

and denote its expectations under different distributions as:

$$\mathcal{R}_0^{(M)}(\theta) = \mathbb{E}_{(\boldsymbol{X},y)\sim p}\mathcal{L}_0^{(M)}(\theta)$$
$$\hat{\mathcal{R}}_0^{(M)}(\theta) = \mathbb{E}_{(\boldsymbol{X},\hat{y})\sim p_\gamma}\mathcal{L}_0^{(M)}(\theta). \tag{28}$$

Let θ_0^* and $\hat{\theta}_0$ be the minimizers of the $\mathcal{R}_0^{(M)}(\theta)$ and $\hat{\mathcal{R}}_0^{(M)}(\theta)$, respectively. Then, we have the following theorem.

Theorem 3.2 (Robustness Guarantee): Under similar regularity conditions as Theorem 3.1, our meta-learning method achieves:

$$\mathcal{R}^{(M)}(\hat{\theta}) - \mathcal{R}^{(M)}(\theta^*) \leq \mathcal{R}_0^{(M)}(\hat{\theta_0}) - \mathcal{R}_0^{(M)}(\theta_0^*) + \mathcal{O}(\eta^2). \tag{29}$$

This theorem establishes that our meta-learning approach achieves smaller risk difference compared to traditional methods, demonstrating its superior robustness in handling noisy pseudo-labels during domain adaptation.

Proof: Following the same noisy label expansion as in Theorem 3.1, we compare our meta-learning loss with the baseline loss $\mathcal{L}_0^{(M)}$. From $\hat{\mathcal{R}}^{(M)}(\hat{\theta}) \leq \hat{\mathcal{R}}^{(M)}(\theta^*)$ and $\hat{\mathcal{R}}_0^{(M)}(\hat{\theta_0}) \leq \hat{\mathcal{R}}_0^{(M)}(\hat{\theta})$, we obtain two inequalities bounding the risk differences. The key distinction is that our meta-update term $\mathcal{L}^A(\theta - \eta\nabla\mathcal{L}^D(\theta))$ introduces an additional gradient alignment term via Taylor expansion: $-\eta\nabla\mathcal{L}^D(\theta)^T\nabla\mathcal{L}^A(\theta) + \mathcal{O}(\eta^2)$, which is absent in $\mathcal{L}_0$. Combining these inequalities yields the stated bound. □

This result shows that when using pseudo-labels for training, our meta-learning method will provide smaller mean differences $\mathcal{R}^{(M)}(\hat{\theta}) - \mathcal{R}^{(M)}(\theta^*)$ in the target domain compared to a traditional loss, which implies the robustness of our meta-learning method in domain adaptation.

Discussion: These theoretical results provide formal guarantees for two key aspects of our method: (1) the effectiveness of prioritizing discriminant samples through meta-learning (Theorem 3.1), and (2) enhanced robustness against pseudo-label noise compared to traditional approaches (Theorem 3.2). Theorem 3.1 mathematically supports our intuition that focusing optimization on informative boundary samples, while using anchor samples for stabilization via the meta-objective, leads to more effective adaptation. Theorem 3.2 confirms that the specific structure of our meta-learning update rule inherently mitigates the negative impact of potentially incorrect pseudo-labels, contributing significantly to the overall stability and reliability.

Summarization: Our cross-branch supervision would act as consistency learning to reduce the potential overfitting of overconfident pseudo-labels [54]. We also balance various adaptation tasks (*i.e.*, discriminant samples and anchor samples) within a meta-learning framework. We alternate the optimization of both branches.

IV. Experiments

A. Experimental Settings

Datasets: The experiments of our ROSE are performed on real-world benchmark datasets. For credibility, we executed tests in both *cross-dataset* and *dataset split* source-free domain adaptation. *For cross-dataset experiments*, the dataset is inherently unbiased across sub-datasets. We adapt COX2 [59] and BZR [59] datasets. *For dataset-split experiments*, we follow previous works [60], [61], [62] to split the dataset by graph density, where we conduct experiments on Mutagenicity [63], FRANKENSTEIN [64], PROTEINS [65] and TWITTER-Real-Graph-Partial [66].

Baselines: The details of the baseline methods are:

- **GCN** [67] adopts a localized form of spectral convolution to generate representations for graph data.
- **GIN** [68] uses message-passing graph neural networks to generate representations of the topological structures of graphs.
- **GraphSAGE** [69] utilizes sampling techniques to enhance the efficiency of graph neural networks on large-scale data.
- **GAT** [70] uses an attention mechanism to generate representations that focus on the most informative parts.
- **Mean-Teacher** [71] uses a student model to make predictions and a teacher model to generate training targets, in a semi-supervised manner.
- **InfoGraph** [72] is a semi-supervised GNN method using mutual information maximization for graph representation learning. We adopt the same configuration as ROSE.
- **TGNN** [49] is a semi-supervised graph classifier combining message-passing and graph-kernel modules. We adopt 16 hidden graphs and 0.5 temperature.
- **SHOT** [73] is an SFDA method that freezes the source classifier and fine-tunes the feature extractor using pseudo-labels and information maximization. For fair comparison, we employ GCN as the encoder, adhering to the recommended hyperparameters.
- **PLUE** [13] is a state-of-the-art SFDA method that evaluates label reliability. We set the Temporal Queue Length to 5 as recommended and adapt it for graph-level classification using a GCN encoder.
- **TAST** [74] is an SFDA method that utilizes nearest neighbor information for target domain self-training. We set the number of gradient steps per adaptation to 2.
- **CoCo** [75] is a graph-level UDA method that learns domain-invariant representations by maximizing mutual information between source and target domains. We use GCN as the encoder, and follow the original setup.
- **FRGNN** [46] is a SFGDA method that reconstructs node features using an output-to-input mapping from the pre-trained GNN to mitigate distribution shift at test time without retraining. We set the pooling ratio to 0.4 and utilize GCN for graph feature encoding.
- **GALA** [44] is a recent SFGDA method using graph diffusion for graph reconstruction, using class-specific thresholding, and graph jigsaw for domain adaptation. We use GCN as the encoder, and follow the original setup.

Implementation Details: We initiate the baseline methods with the hyperparameters suggested by the original publications and fine-tune them to optimize the models. In order to address the effects of randomness, we employed 5 runs and reported the average accuracy and standard deviation. In ROSE, we select GCN as the default GNN encoder. Our configuration entails an embedding dimension of 128 and includes two layers for processing. We employ the Adam optimizer with an initial learning rate of 0.01 and a batch size of 128. Before domain adaptation, the graph-kernel branch $\mathcal{L}^{(K)}$ undergoes an initial training phase of 20 epochs on pseudo-labels that are generated by the message-passing branch.

In the adaptation process, we first generate pseudo-labels from the model trained on the source data. We select the top 60%

TABLE I
THE CLASSIFICATION RESULTS (IN %) ON MUTAGENICITY (SOURCE → TARGET). M0, M1, M2, AND M3 ARE THE SUB-DATASETS

Methods	M0→M1	M1→M0	M0→M2	M2→M0	M0→M3	M3→M0	M1→M2	M2→M1	M1→M3	M3→M1	M2→M3	M3→M2	Avg.
GCN	68.0±2.0	68.8±1.5	60.5±2.8	64.4±1.5	53.7±1.3	58.1±1.4	75.2±0.8	76.2±1.5	67.5±1.2	55.4±1.5	62.5±1.0	68.5±1.5	64.9±1.5
GIN	70.6±0.4	64.2±0.9	63.5±1.2	62.5±0.7	57.0±0.1	56.4±0.3	73.3±0.5	76.5±0.4	65.2±0.6	53.3±1.8	64.4±0.8	66.8±0.5	64.5±0.7
GraphSAGE	71.2±0.6	65.6±0.3	64.3±0.3	65.5±0.2	57.3±0.5	56.5±0.3	74.7±0.4	77.6±0.7	62.3±0.4	51.7±0.3	62.4±0.5	62.3±0.7	64.3±0.4
GAT	69.7±1.0	67.0±1.6	62.7±2.3	67.0±1.5	56.1±2.1	57.8±1.7	76.6±1.2	77.2±0.4	63.4±1.3	53.0±3.9	60.7±0.6	61.8±3.2	64.6±1.7
Mean-Teacher	65.3±4.7	52.1±3.4	70.6±2.5	52.2±1.9	49.9±0.5	49.0±0.4	66.2±1.4	62.7±4.1	50.1±1.3	72.2±1.6	48.9±1.7	65.8±3.1	58.7±2.2
InfoGraph	69.1±1.8	68.9±0.3	66.6±2.5	64.9±1.2	55.9±1.0	57.8±2.1	74.7±0.3	76.8±1.5	65.6±0.6	57.1±3.2	64.7±1.9	64.2±2.9	65.5±1.6
TGNN	73.3±4.9	61.9±2.4	65.3±4.0	58.1±2.4	55.5±3.5	58.1±2.4	65.9±1.1	66.7±3.9	66.5±2.1	70.1±1.0	55.5±3.5	65.3±3.0	63.5±2.9
SHOT	74.4±1.4	64.7±1.6	66.9±2.2	67.2±1.5	61.2±2.1	52.9±1.8	63.0±2.3	72.0±1.7	59.7±2.4	72.0±2.3	60.8±1.9	69.1±2.1	65.3±1.9
PLUE	75.2±1.4	68.5±0.5	66.3±1.1	67.9±1.6	54.0±1.3	56.4±1.4	68.4±1.0	76.9±3.5	62.9±0.4	57.6±3.0	62.0±0.6	67.4±2.6	65.3±1.5
TAST	74.2±1.5	64.0±1.8	70.2±1.3	67.6±1.4	62.9±1.9	57.2±1.7	68.5±1.2	77.2±1.6	60.7±1.5	73.4±1.8	62.4±1.4	69.4±1.7	67.3±1.6
CoCo	73.1±1.3	67.1±1.7	70.2±1.4	68.0±1.6	59.8±2.0	53.7±1.9	65.6±1.5	72.3±1.8	58.9±1.6	75.8±1.7	64.2±1.3	70.8±1.5	66.6±1.6
FRGNN	72.8±1.5	65.5±1.8	68.9±1.4	67.0±1.6	60.5±2.0	53.9±1.9	66.9±1.7	73.5±1.2	59.0±1.5	74.3±1.3	63.1±1.6	70.1±1.4	66.3±1.6
GALA	76.4±0.8	69.6±1.3	70.0±2.2	63.2±1.2	58.4±1.2	60.6±1.3	76.9±1.8	78.1±1.9	65.7±1.1	66.5±3.1	65.6±1.5	70.6±1.3	68.6±1.6
ROSE	76.3±1.3	69.6±1.6	70.8±2.1	68.3±0.9	61.9±2.6	60.8±2.1	71.5±1.2	79.7±1.1	61.6±0.7	76.1±1.5	64.8±1.4	71.3±1.8	**69.4**±1.5

TABLE II
THE CLASSIFICATION RESULTS (IN %) ON PROTEINS (SOURCE → TARGET). P0, P1, P2, AND P3 ARE THE SUB-DATASETS

Methods	P0→P1	P1→P0	P0→P2	P2→P0	P0→P3	P3→P0	P1→P2	P2→P1	P1→P3	P3→P1	P2→P3	P3→P2	Avg.
GCN	71.9±0.9	74.7±2.9	62.6±1.2	68.3±3.8	51.1±3.3	45.8±3.1	57.6±2.1	70.4±2.2	39.7±3.6	49.7±2.6	58.3±1.3	52.9±3.0	58.6±2.5
GIN	70.0±2.1	60.7±3.6	61.8±2.6	72.9±2.7	47.1±3.3	44.3±4.2	62.5±2.1	68.9±2.0	41.1±3.2	47.9±3.3	48.6±4.0	56.1±2.6	57.4±3.0
GraphSAGE	72.2±1.0	78.3±3.0	64.7±2.3	67.2±1.1	46.9±1.1	42.2±4.0	62.6±1.8	69.7±0.8	32.9±2.0	50.8±2.3	56.1±3.5	56.9±3.4	58.9±2.2
GAT	70.0±3.7	71.4±3.7	66.8±2.0	73.9±2.6	49.3±1.3	40.4±2.8	61.4±4.6	68.9±2.2	44.3±3.8	49.3±3.6	50.7±2.4	52.1±2.9	58.3±3.0
Mean-Teacher	64.3±4.1	71.4±5.2	60.4±3.6	72.1±4.4	25.0±5.6	55.4±4.0	61.1±2.3	60.7±5.3	29.6±4.8	49.3±3.1	31.8±4.0	55.4±4.9	51.0±4.3
InfoGraph	74.0±2.7	77.6±2.9	68.3±3.6	71.1±1.1	46.9±3.2	46.5±2.1	64.4±1.5	72.2±1.9	41.9±1.1	35.4±1.9	54.7±3.1	62.6±4.2	59.4±2.4
TGNN	60.4±3.4	41.8±2.8	45.7±4.6	46.6±4.2	64.8±5.4	42.0±3.9	58.6±3.2	58.4±3.7	75.7±4.0	68.4±3.2	65.5±4.8	53.8±2.0	55.9±3.8
SHOT	54.5±2.3	73.2±1.8	67.5±2.1	69.6±1.9	35.7±3.2	57.8±2.4	68.9±1.7	63.7±2.2	39.3±3.1	49.3±2.8	57.1±2.5	66.7±1.9	58.6±2.3
PLUE	64.3±2.3	71.5±2.1	65.0±3.0	78.4±2.1	45.9±3.7	63.7±3.0	58.3±3.3	68.9±2.7	41.9±1.3	45.7±4.1	47.5±3.2	58.3±2.7	59.1±2.8
TAST	58.4±2.1	76.7±1.6	72.8±1.8	69.8±2.2	45.7±2.9	63.9±2.3	67.5±1.9	69.6±1.7	35.7±3.4	62.5±2.5	57.1±2.8	67.8±1.8	62.3±2.3
CoCo	59.9±1.9	74.8±1.7	70.9±2.0	70.9±1.8	44.0±3.0	63.9±2.2	68.5±1.8	63.1±2.1	43.6±2.9	51.4±2.6	57.7±2.4	69.1±1.7	61.5±2.2
FRGNN	60.3±1.8	74.2±1.7	72.8±1.5	70.3±2.0	44.9±2.5	61.2±2.1	66.9±1.6	62.6±1.9	42.9±2.2	50.3±2.4	57.5±1.8	68.0±2.0	61.0±2.0
GALA	72.3±1.0	73.9±2.0	66.8±3.4	78.5±1.9	47.1±2.2	65.6±3.7	61.1±1.1	71.8±3.1	43.9±2.0	50.4±2.6	43.2±2.0	60.0±3.4	61.2±2.4
ROSE	66.8±1.4	78.6±1.6	76.1±0.8	75.6±1.4	46.1±2.6	64.3±4.2	69.7±2.6	64.6±2.1	45.4±2.3	51.4±3.1	58.6±4.5	69.3±2.4	**63.9**±2.4

TABLE III
THE CLASSIFICATION RESULTS (IN %) ON FRANKENSTEIN (SOURCE → TARGET). F0, F1, F2, AND F3 ARE THE SUB-DATASETS

Methods	F0→F1	F1→F0	F0→F2	F2→F0	F0→F3	F3→F0	F1→F2	F2→F1	F1→F3	F3→F1	F2→F3	F3→F2	Avg.
GCN	55.3±0.8	56.4±1.6	60.4±1.9	54.6±1.4	46.7±1.8	51.6±1.7	60.7±0.8	58.3±1.3	47.9±1.8	47.5±0.8	52.1±2.6	54.6±1.8	53.8±1.5
GIN	56.5±0.5	53.8±1.4	57.2±0.4	57.9±1.8	50.7±3.5	51.0±0.6	58.7±1.3	58.3±0.8	46.7±1.4	47.9±1.3	50.5±0.7	52.0±2.1	53.4±1.3
GraphSAGE	56.6±1.3	53.5±0.4	55.4±1.0	57.8±1.3	49.9±0.3	55.5±1.6	59.4±1.4	59.6±0.2	47.1±1.0	49.3±0.8	49.2±1.7	52.7±1.3	53.8±1.0
GAT	57.1±0.9	56.0±0.8	58.5±1.8	56.4±2.2	48.7±1.1	51.6±2.8	62.5±1.8	57.1±0.7	46.2±2.2	47.2±1.4	51.7±2.6	53.6±0.9	53.9±1.6
Mean-Teacher	57.0±4.6	53.8±3.3	55.6±3.5	54.2±2.5	47.6±3.8	49.7±1.5	56.2±3.4	59.1±4.8	48.5±3.4	52.9±5.2	51.4±3.3	53.1±4.7	53.2±3.7
InfoGraph	57.0±2.7	55.7±2.2	60.1±2.6	60.0±3.0	48.9±2.0	51.2±1.7	60.6±1.1	61.8±1.9	45.4±2.3	46.3±1.2	53.2±2.0	53.5±0.8	54.5±1.9
TGNN	53.2±5.6	48.8±1.7	54.0±5.2	54.2±1.7	46.2±4.0	47.9±0.5	55.1±5.1	50.8±3.6	56.8±4.0	53.2±5.6	56.8±3.0	48.1±4.5	53.1±3.7
PLUE	57.9±1.2	56.4±1.3	60.0±1.9	59.1±1.6	49.1±0.6	53.2±1.8	60.8±1.5	52.3±3.7	48.1±3.7	52.1±4.1	52.7±1.5	53.9±2.2	54.6±2.1
SHOT	62.1±1.4	54.2±2.6	57.0±1.2	59.1±1.5	50.7±2.1	52.2±1.8	60.8±3.3	61.8±1.7	51.6±3.4	53.5±2.9	52.8±1.7	56.2±2.6	56.0±2.2
TAST	63.2±2.5	54.4±1.8	56.2±2.3	58.5±3.4	51.6±3.9	52.3±1.7	61.7±2.2	60.5±1.6	55.7±1.5	54.1±2.8	52.5±1.9	56.2±1.8	56.4±2.3
CoCo	64.7±1.3	54.6±1.7	58.6±2.4	52.4±1.6	50.8±2.3	51.0±2.9	57.7±1.6	67.8±2.8	55.1±2.6	57.6±1.7	52.2±1.8	53.0±2.5	56.3±2.1
FRGNN	63.8±1.5	53.2±1.8	56.7±1.4	53.9±1.6	50.2±2.0	51.7±1.9	60.2±1.7	62.9±1.2	53.3±1.5	56.1±1.3	51.8±1.6	54.5±1.4	55.7±1.7
GALA	59.7±0.6	56.9±0.9	58.8±1.1	53.8±0.4	51.7±1.2	55.9±1.7	61.0±0.5	62.7±0.7	53.0±0.6	54.5±1.0	53.5±0.5	56.4±0.8	56.5±0.8
ROSE	66.7±1.2	56.4±2.1	61.1±1.5	61.0±1.3	51.5±2.1	52.9±2.3	63.4±0.6	65.7±3.4	54.3±0.9	60.3±2.6	53.0±0.8	56.7±1.3	**58.6**±1.7

of data with the highest confidence scores as reliable pseudo-labeled data. The adaptation process is conducted over 20 epochs using the Adam optimizer.

For hyperparameters, the default discriminant set ratio is set at 30%, and the default loss balance scalar α is 0.5. These parameters will be validated in Section IV-G. All experiments are conducted on NVIDIA A40 GPUs.

B. Performance Comparison

Observations: Table I, II, III, IV and V presents the comparative performance of various methods.

The following observations can be made: (1) SFGDA presents substantial challenges. The significant performance degradation calls for addressing this practical issue. The

TABLE IV
THE CLASSIFICATION RESULTS (IN %) ON TWITTER-REAL-GRAPH-PARTIAL (SOURCE → TARGET). T0, T1, T2, AND T3 ARE SUB-DATASETS

Methods	T0→T1	T0→T2	T0→T3	T1→T0	T1→T2	T1→T3	T2→T0	T2→T1	T2→T3	T3→T0	T3→T1	T3→T2	Avg.
GIN	57.6±0.3	60.8±0.5	58.5±1.1	61.3±0.2	61.1±0.3	59.3±0.8	59.7±0.6	61.2±0.5	59.0±1.3	60.3±0.6	59.8±0.8	60.4±1.6	59.9±0.7
GCN	60.2±0.4	60.7±0.5	58.2±1.2	61.1±0.9	61.3±0.5	59.8±0.7	60.2±0.7	61.8±0.6	59.5±1.5	60.7±0.7	59.4±1.7	60.7±1.2	60.3±0.9
GAT	58.3±0.4	61.2±0.7	59.0±1.0	60.1±0.7	61.6±0.6	58.8±0.6	60.5±0.7	60.6±0.5	59.5±1.2	60.4±0.7	59.1±1.4	59.7±1.9	59.9±0.9
GraphSAGE	59.0±0.5	60.5±0.6	60.2±0.9	59.9±1.0	60.8±0.5	60.8±0.7	60.2±0.3	60.4±0.4	58.9±1.0	59.7±0.4	60.2±1.3	60.4±1.5	60.1±0.8
MeanTeacher	54.0±1.2	50.2±1.3	51.1±2.0	51.0±0.9	50.6±0.7	48.1±4.5	52.7±0.4	51.7±1.9	52.6±1.7	52.1±1.0	51.0±3.5	51.1±1.8	51.4±1.7
InfoGraph	61.7±1.3	62.1±0.9	60.5±1.0	60.6±0.9	61.0±0.3	59.2±1.1	62.3±1.0	61.3±0.7	60.8±1.8	61.3±1.1	62.4±1.9	61.3±1.7	61.2±1.1
TGNN	58.4±0.6	60.0±1.0	58.0±1.5	59.0±0.8	59.4±1.3	57.4±0.7	57.7±0.9	58.1±1.1	58.4±1.7	58.2±0.9	57.9±1.4	60.4±1.2	58.6±1.1
SHOT	61.1±1.4	61.3±1.6	60.2±1.2	60.4±1.5	59.3±2.1	60.5±1.8	60.7±1.3	61.8±1.7	58.9±1.4	60.0±1.9	60.6±1.5	62.1±1.6	60.6±1.6
PLUE	58.6±1.0	59.5±0.9	55.5±0.9	57.7±0.7	58.1±0.8	56.1±1.2	56.3±0.8	57.2±0.9	57.8±1.8	57.7±1.0	57.6±1.9	60.2±2.2	57.7±1.2
TAST	61.2±1.5	61.5±1.8	60.8±1.3	60.7±1.4	58.6±1.9	61.1±1.7	60.6±1.2	61.1±1.6	59.5±1.5	60.7±1.8	60.8±1.4	61.3±1.7	60.7±1.6
CoCo	61.7±1.3	62.0±1.7	61.7±1.4	60.5±1.6	62.1±2.0	60.8±1.9	62.6±1.5	63.5±1.8	59.6±1.6	60.5±1.7	61.1±1.3	61.5±1.5	61.5±1.6
FRGNN	61.0±1.5	61.2±1.6	61.2±1.4	60.3±1.7	60.3±1.3	60.1±1.8	61.8±1.5	62.9±1.6	60.2±1.4	60.8±1.7	61.0±1.3	61.1±1.8	61.0±1.6
GALA	62.1±0.8	62.0±1.3	61.8±2.2	60.9±1.2	61.9±1.2	60.5±1.3	62.3±1.8	63.1±1.9	60.5±1.1	61.2±0.9	61.1±1.1	61.3±1.3	61.6±1.3
ROSE	63.2±0.7	62.7±1.0	62.1±1.1	61.7±0.6	63.2±0.7	61.2±1.3	62.9±0.8	63.2±1.0	61.4±0.7	61.9±0.9	61.4±1.1	62.4±1.6	**62.3**±0.9

TABLE V
THE CLASSIFICATION RESULTS (IN %) ON COX2 AND BZR (SOURCE → TARGET)

Methods	C→CM	CM→C	B→BM	BM→B	Avg.
GCN	54.1±2.6	46.6±4.1	51.3±2.3	62.8±3.7	53.7±3.2
GIN	51.1±2.2	46.4±4.6	48.1±3.6	65.6±2.8	52.8±3.3
GraphSAGE	49.2±3.4	42.9±3.9	47.3±1.5	67.5±3.1	51.7±3.0
GAT	52.0±1.8	48.9±3.7	48.4±2.2	61.3±4.2	52.6±3.0
Mean-Teacher	53.0±2.3	42.6±4.9	50.6±2.1	57.6±4.3	50.9±3.4
InfoGraph	45.9±3.4	48.9±3.3	51.9±3.2	65.2±4.7	53.4±3.6
TGNN	48.3±4.2	52.1±5.6	46.4±2.7	68.5±4.3	53.8±4.2
SHOT	51.6±2.8	70.5±3.9	56.7±2.5	49.9±3.4	57.2±3.2
PLUE	54.4±1.4	41.7±3.2	49.8±2.8	74.8±1.5	55.2±2.2
TAST	52.1±2.4	72.2±3.6	56.2±2.9	52.1±3.8	58.2±3.2
CoCo	56.1±2.7	67.1±3.5	52.4±2.6	72.4±3.2	62.0±3.0
FRGNN	56.4±2.1	59.6±3.0	51.0±2.5	65.3±3.5	58.1±2.8
GALA	56.6±0.7	59.1±2.6	53.2±2.0	73.2±3.1	60.5±2.1
ROSE	57.4±1.8	67.8±4.3	52.5±3.6	73.6±2.8	**62.8**±3.1

TABLE VI
ABLATION STUDIES. M, P, F, C, AND B ARE FOR MUTAGENICITY, PROTEINS, FRANKENSTEIN, COX2, AND BZR

	M	P	F	C	B	Avg.
V1	68.1±1.8	63.1±2.4	56.3±1.9	60.3±4.0	62.3±3.1	62.0±2.6
V2	68.4±1.7	63.5±3.2	56.8±1.4	61.0±3.7	62.2±3.2	62.4±2.6
V3	69.0±2.5	63.4±2.1	57.6±1.6	62.1±3.2	62.6±1.6	62.9±2.2
V4	69.3±2.6	63.8±2.9	57.9±2.5	62.7±2.6	63.1±2.8	63.4±2.7
V5	68.9±2.6	64.0±2.8	58.4±2.1	62.8±2.4	62.9±4.1	63.4±2.8
V6	69.6±2..0	63.8±3.2	58.6±2.0	61.6±3.3	63.7±2.7	**63.5**±2.6
ROSE	69.4±1.5	63.9±2.4	58.6±1.7	62.6±3.1	63.1±3.2	**63.5**±2.4

TABLE VII
CROSS-DATASET ADAPTATION PERFORMANCE (IN %) BETWEEN FRANKENSTEIN AND PROTEINS

Task	Direct Test	After Adaptation
PROTEINS → FRANKENSTEIN	43.20	**48.18%**
FRANKENSTEIN → PROTEINS	47.98	**51.76%**

ineffectiveness of the baseline methods suggests that previous work cannot handle the realistic yet complicated scenarios effectively. (2) Semi-supervised approaches (e.g., InfoGraph), generally surpass source-only methods. These approaches use both the labeled source data and unlabeled target data. However, access to source data is difficult and even impossible in the real world. Despite this, due to a lack of consideration for domain shift, semi-supervised methods suffer from poor stability and degraded performance. (3) SFDA methods (*i.e.*, SHOT, PLUE, TAST) perform better than baselines (as Table I, II). While they represent the SOTA SFDA performance for image classification tasks, it is important to note that they were not specifically devised for graph data or significant domain shifts. (4) Graph UDA methods (*i.e.*, CoCo) achieve competitive performance, but it's worth noting that they utilize both source and target domain graphs, which are usually difficult to acquire in real-world scenarios. (5) ROSE demonstrates marked improvements in both split sub-datasets and cross-dataset scenarios, particularly in the case where other methods perform poorly.

Cross-Dataset Adaptation: To further validate the generalization capability of ROSE across significantly different datasets, we conducted cross-dataset experiments between FRANKENSTEIN (molecular compound graphs) and PROTEINS (protein structure graphs). Table VII presents the results. Despite substantial structural differences between these datasets (PROTEINS graphs are 2.3× larger in node count and 4× larger in edge count), ROSE achieves consistent improvements in both directions. Specifically, in the PROTEINS → FRANKENSTEIN direction, accuracy improves from 43.20% to 48.18%, while in the FRANKENSTEIN → PROTEINS direction, accuracy increases from 47.98% to 51.76%. These results demonstrate that ROSE can effectively perform source-free adaptation even across datasets with significant domain gaps, validating the robustness of our dual-branch architecture and neighborhood mining strategy.

Discussion: The enhancements of ROSE can be ascribed to: (1) The cross supervision between branches has improved the model's robustness in representation, particularly in scenarios where labels are scarce (e.g., Table I M0→M3, M3→M0), providing complementary representations from diverse

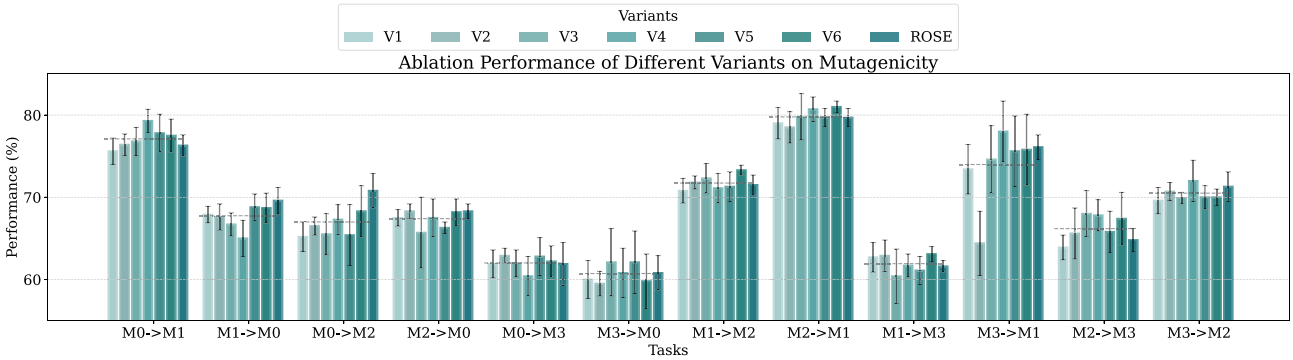


Fig. 3. The ablation results (in %) on Mutagenicity (source → target). M0, M1, M2, and M3 are split by the graph density.

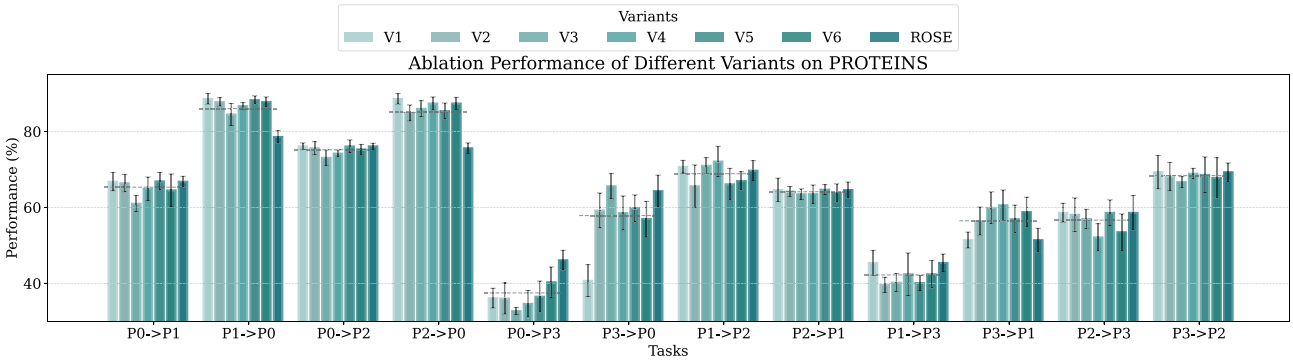


Fig. 4. The ablation results (in %) on PROTEINS (source → target). P0, P1, P2, and P3 are split by the graph density.

perspectives. (2) Neighborhood mining has improved the effectiveness of data utilization in the target domain. This mechanism effectively leverages the relational information within the data to address the lack of explicit labels. (3) Robust cross supervision with meta-learning has increased the stability of the model throughout the adaptation process.

C. Ablation Study

We introduce ablation variants as follows and present results in Table VI, where datasets are denoted by the first letter: *(1) V1*, which excludes the cross-supervision with graph-kernel branch from the full model. *(2) V2*, which replaces the graph-kernel branch with the messaging branches from the full model, applies cross-supervision and initializes the newly introduced modules with pseudo-labels. *(3) V3*, which removes the neighborhood mining for discriminant graphs with meta-learning based on the full model. *(4) V4, V5, and V6* replace the backbone with GIN, GraphSage and GAT, respectively.

Analysis: Ablation results (Table VI, Fig. 3) confirm component effectiveness. (1) Removing cross-supervision (*V1*) significantly drops performance, proving the value of complementary views for robustness. (2) Using two Message-Passing branches (*V2*) is worse than Message-Passing+Graph-Kernel, highlighting the benefit of the Graph-Kernel branch's distinct topological perspective. (3) Removing neighborhood mining and meta-learning (*V3*) degrades performance, showing this mechanism is vital for stable adaptation by balancing informative and stable samples. (4) Strong results with GIN, GraphSAGE, and GAT (*V4-V6*) demonstrate backbone robustness. Overall, ROSE's components synergistically achieve superior SFGDA results.

D. Visualization

From Fig. 6, The visual representations reveal that our algorithm is more adept at capturing potentially ambiguous and boundary-located graph data, which are rich in information. Engaging with these discriminant graph samples for pseudo-label learning allows for deeper mining of the information latent in the unlabeled target domain data, thereby boosting the effectiveness of the domain adaptation. By balancing the learning between the discriminant and anchor sets, our model achieves more stable domain adaptation.

E. Time Efficiency and Scalability

Fig. 8 presents the time efficiency and scalability analysis across datasets with varying node counts. Our method demonstrates superior computational efficiency compared to most baselines, maintaining consistently low execution times across all scales. While InfoGraph and TAST show significant performance degradation with increasing graph complexity, our

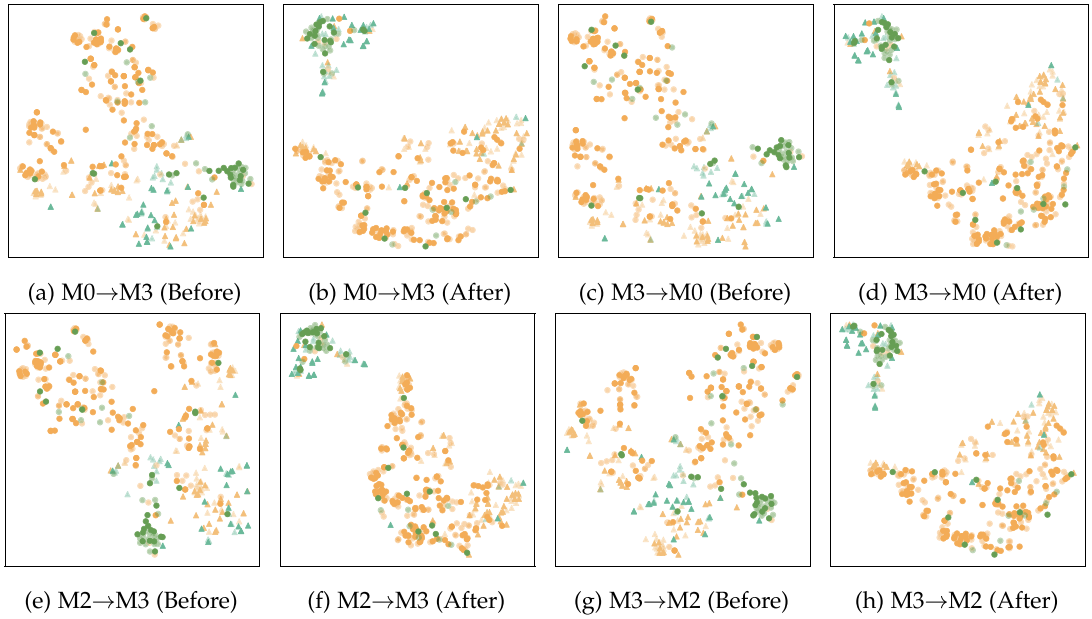


Fig. 5. Visualization of Mutagenicity using t-SNE. Triangles (△) symbolize the discriminant set, while circles (●) represent the anchor set. Discriminant samples, identified by high neighborhood entropy, tend to lie near classification boundaries, providing rich information crucial for adapting the decision boundary. Anchor samples, representing more stable, core class examples, provide consistent learning signals that help stabilize the adaptation process. The plots show the shift in data distribution before and after adaptation, highlighting the effectiveness of the adaptation process.

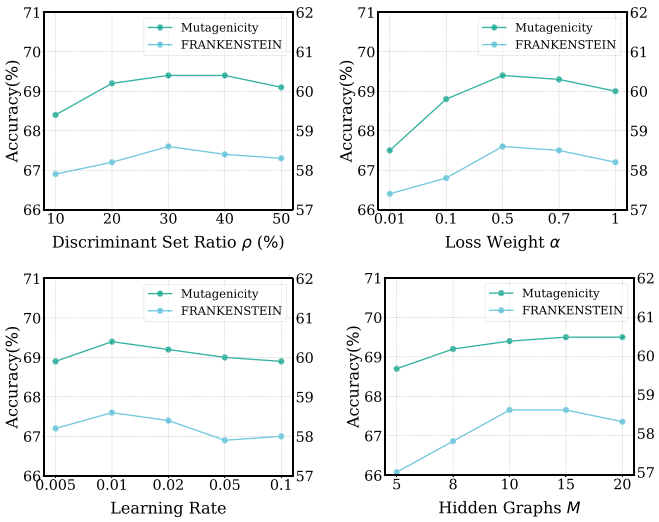


Fig. 6. Sensitivity analysis of key hyperparameters.

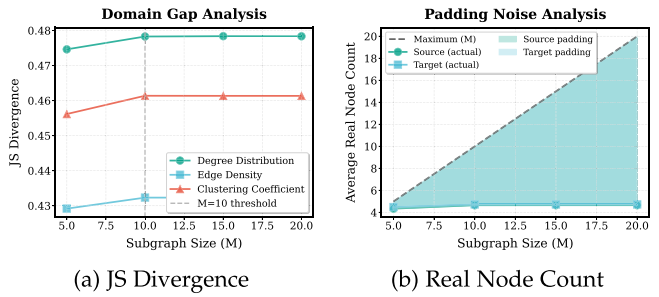


Fig. 7. Analysis of subgraph size M impact. (a) JS divergence stabilizes at $M = 10$; (b) real node count growth is limited, indicating padding dominates for large M.

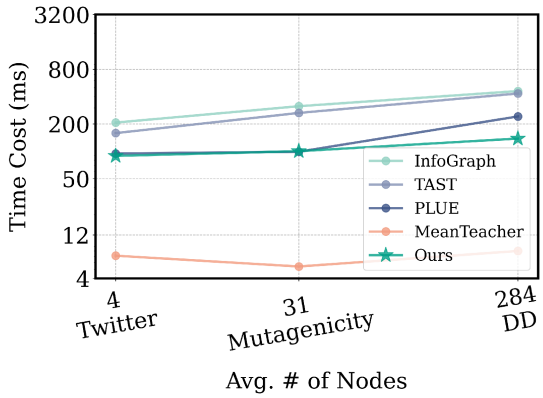


Fig. 8. Time efficiency and scalability analysis on ROSE with baseline methods.

approach maintains efficiency. These results confirm that our approach achieves state-of-the-art performance while maintaining practical computational efficiency for real-world deployment scenarios.

F. Convergence Analysis

Fig. 9 illustrates the convergence behavior of ROSE compared to TAST. The results indicate that ROSE typically converges faster and achieves lower loss values in the target domain, highlighting the effectiveness of its meta-learning optimization and cross-branch supervision components.

G. Sensitivity Analysis

Impact of Discriminant Set Ratio ρ: Fig. 6 (top left) shows accuracy peaks at $\rho = 30\%$. This suggests an optimal balance

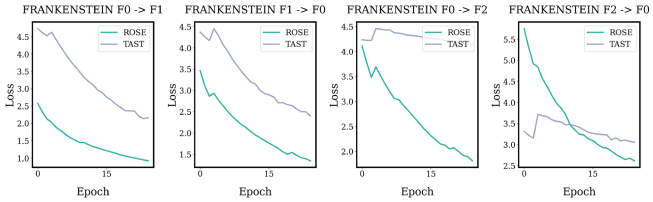


Fig. 9. Convergence analysis on ROSE with TAST.

TABLE VIII
ROBUSTNESS ANALYSIS OF ROSE WITH DIFFERENT SOURCE MODEL CHECKPOINTS ON THE MUTAGENICITY (M0→M3 TASK)

Methods	Ckpt A	Ckpt B	Ckpt C	Ckpt D
Base Model	53.8	53.6	52.9	53.2
ROSE	**58.6**	**57.9**	**57.9**	**58.1**

between leveraging informative boundary samples (lower ρ) and avoiding excessive noise or less informative data (higher ρ). The default is set to $\rho = 30\%$.

Impact of Loss Balance Weight α: Fig. 6 (top right) indicates peak performance around $\alpha = 0.5$. This underscores the importance of balancing the meta-learning updates derived from the discriminant and anchor sets (Eq. 17, 18) for stable adaptation. The default is $\alpha = 0.5$.

Impact of Learning Rate: Fig. 6 (bottom left) suggests optimal performance near a learning rate of 0.01 during sensitivity testing. Lower rates risk slow convergence, while higher rates can destabilize training. We adopt a default adaptation learning rate of 0.01 for robust performance.

Impact of Hidden Graphs M: Fig. 6 (bottom right) shows accuracy generally increasing up to $M = 10$, then stabilizing or declining slightly. To understand this phenomenon, we conducted subgraph structure analysis on PROTEINS dataset (P0→P3) with $M \in [5, 10, 15, 20]$. Fig. 7 reveals two key findings: (a) JS divergence between source and target domain subgraphs increases from $M = 5$ to $M = 10$ and then stabilizes, indicating $M = 10$ already captures maximum structural discrepancy; (b) actual node count does not scale linearly with M, showing that increasing M beyond 10 primarily adds padding noise rather than capturing more real structural information. These findings suggest that $M > 10$ leads to overfitting source-specific structures and cumulative padding noise, while $M = 10$ achieves optimal balance between capturing structural information and avoiding overfitting. The default is $M = 10$.

Robustness to Source Model Checkpoints: To assess the robustness of ROSE to variations in the pre-trained source model, we tested its performance using different source model checkpoints, as detailed in Table VIII. The results consistently show that ROSE provides significant improvements over the base model across all tested checkpoints, demonstrating its robustness and effectiveness.

V. CONCLUSION

In this work, we present a novel solution for source-free graph domain adaptation, addressing the shortage of labeled data in knowledge transfer. Utilizing a dual-branch architecture, ROSE harnesses both implicit and explicit graph topological information. It incorporates neighborhood mining and meta-learning to enhance optimization stability under label scarcity and utilizes cross-branch supervision to improve robustness. Extensive experiments demonstrate ROSE's superior performance over existing baselines, validating its effectiveness without requiring source-domain samples.

Limitations: ROSE has brought progress in the field of source-free domain adaptation, but it still has certain limitations. In real-world applications, target samples may come from unseen classes, where ROSE cannot handle this open-set scenario currently. In the future, we will explore extending ROSE to more generalized real-world scenarios to address this limitation and improve its practicality. Future work will also focus on developing more efficient adaptation strategies, such as graph sampling and sparsification, to further reduce computational cost and enhance scalability.

REFERENCES

[1] C. Lu, Q. Liu, C. Wang, Z. Huang, P. Lin, and L. He, "Molecular property prediction: A multilevel quantum interactions modeling perspective," in *Proc. Assoc. Advance. Artif. Intell.*, 2019, pp. 1052–1060.
[2] H. Xu, C. Jiang, X. Liang, and Z. Li, "Spatial-aware graph relation network for large-scale object detection," in *Proc. Comput. Vis. Pattern Recognit.*, 2019, pp. 9298–9307.
[3] Y. Zeng et al., "Keyword-based diverse image retrieval with variational multiple instance graph," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 12, pp. 10528–10537, Dec. 2023.
[4] X. Ma et al., "Fused Gromov-Wasserstein graph mixup for graph-level classifications," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 15252–15276.
[5] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017.
[6] Q. Sun et al., "Sugar: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism," in *Proc. ACM Web Conf.*, 2021, pp. 2081–2091.
[7] J. Luo et al., "Sparse causal discovery with generative intervention for unsupervised graph domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2025.
[8] M. Lin, W. Li, D. Li, Y. Chen, G. Li, and S. Lu, "Multi-domain generalized graph meta learning," in *Proc. Assoc. Advance. Conf. Artif. Intell.*, 2023, pp. 4479–4487.
[9] Y. Sui et al., "Unleashing the power of graph data augmentation on covariate distribution shift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 18109–18131.
[10] C. Chen, L. Tang, Y. Huang, X. Han, and Y. Yu, "Coda: Generalizing to open and unseen domains with compaction and disambiguation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2023, vol. 36, pp. 12746–12759.
[11] Z. Li, R. Cai, G. Chen, B. Sun, Z. Hao, and K. Zhang, "Subspace identification for multi-source domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 34504–34518.
[12] Y. Sun, X. Wang, Z. Liu, J. Miller, A. Efros, and M. Hardt, "Test-time training with self-supervision for generalization under distribution shifts," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9229–9248.
[13] M. Litrico, A. Del Bue, and P. Morerio, "Guiding pseudo-labels with uncertainty estimation for source-free unsupervised domain adaptation," in *Proc. Comput. Vis. Pattern Recognit.*, 2023, pp. 7640–7650.
[14] P. Velickovic, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, "Deep graph infomax," in *Proc. Int. Conf. Learn. Representations*, 2019.
[15] K. Huang and M. Zitnik, "Graph meta learning via local subgraphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 5862–5874.
[16] S. Gui, X. Li, L. Wang, and S. Ji, "Good: A graph out-of-distribution benchmark," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 2059–2073.
[17] Z. Xiao et al., "SPA: A graph spectral alignment perspective for domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 37252–37272.

[18] D. Xia, X. Wang, N. Liu, and C. Shi, "Learning invariant representations of graph neural networks via cluster generalization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 45602–45613.

[19] X. Chen, R. Cai, Y. Fang, M. Wu, Z. Li, and Z. Hao, "Motif graph neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 10, pp. 14833–14847, Oct. 2024.

[20] J. Yu, J. Liang, and R. He, "Mind the label shift of augmentation-based graph OOD generalization," in *Proc. Comput. Vis. Pattern Recognit.*, 2023, pp. 11620–11630.

[21] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Learning adaptive neighborhoods for graph neural networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 22541–22550.

[22] K. Chen et al., "Distribution knowledge embedding for graph pooling," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 7898–7908, Aug. 2023.

[23] P. Wang, J. Luo, Y. Shen, M. Zhang, S. Heng, and X. Luo, "A comprehensive graph pooling benchmark: Effectiveness, robustness and generalizability," 2024, *arXiv:2406.09031*.

[24] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?," in *Proc. Int. Conf. Learn. Representations*, 2019.

[25] X. Fan, M. Gong, Y. Wu, A. K. Qin, and Y. Xie, "Propagation enhanced neural message passing for graph representation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1952–1964, Feb. 2023.

[26] K. Chen et al., "Improving expressivity of GNNs with subgraph-specific factor embedded normalization," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 237–249.

[27] Y. Wang, S. Liu, T. Zheng, K. Chen, and M. Song, "Unveiling global interactive patterns across graphs: Towards interpretable graph neural networks," in *Proc. 30th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2024, pp. 3277–3288.

[28] U. Kang, H. Tong, and J. Sun, "Fast random walk graph kernel," in *Proc. 2012 SIAM Int. Conf. Data Mining*, 2012, pp. 828–838.

[29] L. Cosmo, G. Minello, M. Bronstein, E. Rodolà, L. Rossi, and A. Torsello, "Graph kernel neural networks," in *Proc. IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, 2024, pp. 6257–6270.

[30] B. Shi, Y. Wang, F. Guo, B. Xu, H. Shen, and X. Cheng, "Graph domain adaptation: Challenges, progress and prospects," 2024, *arXiv:2402.00904*.

[31] Z. Zhang et al., "Divide and contrast: Source-free domain adaptation via adaptive contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 5137–5149.

[32] M. J. Mirza, J. Micorek, H. Possegger, and H. Bischof, "The norm must go on: Dynamic unsupervised domain adaptation by normalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 14765–14775.

[33] W. Ju et al., "A survey of data-efficient graph learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2024, pp. 8104–8113.

[34] M. Wu, S. Pan, C. Zhou, X. Chang, and X. Zhu, "Unsupervised domain adaptive graph convolutional networks," in *Proc. ACM Web Conf.*, 2020, pp. 1457–1467.

[35] Q. Wu, H. Zhang, J. Yan, and D. Wipf, "Handling distribution shifts on graphs: An invariance perspective," in *Proc. Int. Conf. Learn. Representations*, 2021.

[36] Y. Fang, P.-T. Yap, W. Lin, H. Zhu, and M. Liu, "Source-free unsupervised domain adaptation: A survey," 2022, *arXiv:2301.00265*.

[37] S. Yang, Y. Wang, J. Van De Weijer, L. Herranz, and S. Jui, "Generalized source-free domain adaptation," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 8978–8987.

[38] X. Ma, Y. Wang, H. Liu, T. Guo, and Y. Wang, "When visual prompt tuning meets source-free domain adaptive semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2024, vol. 36, pp. 6690–6702.

[39] M. Jing, X. Zhen, J. Li, and C. Snoek, "Variational model perturbation for source-free domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 17173–17187.

[40] J. Luo et al., "A survey on efficient large language model training: From data-centric perspectives," in *Proc. 63rd Annu. Meeting Assoc. Comput. Linguistics*, 2025, pp. 30904–30920.

[41] K. Zhang et al., "A survey of deep graph learning under distribution shifts: From graph out-of-distribution generalization to adaptation," 2024, *arXiv:2410.19265*.

[42] J. Luo et al., "Rank and align: Towards effective source-free graph domain adaptation," in *Proc. Int. Joint Conf. Artif. Intell.*, 2024.

[43] H. Mao et al., "Source free graph unsupervised domain adaptation," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2024, pp. 520–528.

[44] J. Luo et al., "GALA: Graph diffusion-based alignment with jigsaw for source-free domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 9038–9051, Dec. 2024.

[45] Z. Ding et al., "SGOOD: Substructure-enhanced graph-level out-of-distribution detection," in *Proc. ACM Int. Conf. Inf. Knowl. Manage.*, 2024, pp. 467–476.

[46] R. Ding, J. Yang, F. Ji, X. Zhong, and L. Xie, "FR-GNN: Mitigating the impact of distribution shift on graph neural networks via test-time feature reconstruction," *IEEE Internet Things J.*, vol. 11, no. 13, pp. 23521–23531, Jul. 2024.

[47] Q. Long, Y. Jin, Y. Wu, and G. Song, "Theoretically improving graph neural networks via anonymous walk graph kernels," in *Proc. Web Conf.*, 2021, pp. 1204–1214.

[48] T. Schulz, P. Welke, and S. Wrobel, "Graph filtration kernels," in *Proc. Assoc. Advance. Artif. Intell.*, 2022, pp. 8196–8203.

[49] W. Ju et al., "TGNN: A joint semi-supervised framework for graph-level classification," in *Proc. Int. Joint Conf. Artif. Intell.*, 2022.

[50] G. Nikolentzos and M. Vazirgiannis, "Random walk graph neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, vol. 33, pp. 16211–16222.

[51] A. Feng, C. You, S. Wang, and L. Tassiulas, "KerGNNs: Interpretable graph neural networks with graph kernels," in *Proc. AAAI Conf. Artif. Intell.*, 2022, pp. 6614–6622.

[52] F. Wang, Z. Han, Z. Zhang, R. He, and Y. Yin, "MHPL: Minimum happy points learning for active source free domain adaptation," in *Proc. Comput. Vis. Pattern Recognit.*, 2023, pp. 20008–20018.

[53] S. Yang, Y. Wang, J. van de Weijer, L. Herranz, S. Jui, and J. Yang, "Trust your good friends: Source-free domain adaptation by reciprocal neighborhood clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 12, pp. 15883–15895, Dec. 2023.

[54] X. Chen, Y. Yuan, G. Zeng, and J. Wang, "Semi-supervised semantic segmentation with cross pseudo supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2613–2622.

[55] B. Chen, J. Jiang, X. Wang, P. Wan, J. Wang, and M. Long, "Debiased self-training for semi-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, vol. 35, pp. 32424–32437 .

[56] W. Zhang, L. Shen, and C.-S. Foo, "Rethinking the role of pre-trained networks in source-free domain adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 18841–18851.

[57] C. Finn, A. Rajeswaran, S. Kakade, and S. Levine, "Online meta-learning," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 1920–1930.

[58] G. Patel, K. R. Mopuri, and Q. Qiu, "Learning to retain while acquiring: Combating distribution-shift in adversarial data-free knowledge distillation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 7786–7794.

[59] J. J. Sutherland, L. A. O'brien, and D. F. Weaver, "Spline-fitting with a genetic algorithm: A method for developing classification structure-activity relationships," *J. Chem. Inf. Comput. Sci.*, vol. 43, no. 6, pp. 1906–1915, 2003.

[60] M. Ding, J. Tang, and J. Zhang, "Semi-supervised learning on graphs with generative adversarial nets," in *Proc. ACM Int. Conf. Inf. Knowl. Manage.*, 2018, pp. 913–922.

[61] N. Yin et al., "Deal: An unsupervised domain adaptive framework for graph-level classification," in *Proc. ACM Int. Conf. Multimedia*, 2022, pp. 3470–3479.

[62] B. Lu et al., "Graph out-of-distribution generalization with controllable data augmentation," 2023, *arXiv:2308.08344*.

[63] J. Kazius, R. McGuire, and R. Bursi, "Derivation and validation of toxicophores for mutagenicity prediction," *J. Med. Chem.*, vol. 48, no. 1, pp. 312–320, 2005.

[64] F. Orsini, P. Frasconi, and L. De Raedt, "Graph invariant kernels," in *Proc. Int. Joint Conf. Artif. Intell.*, 2015, pp. 3756–3762.

[65] K. M. Borgwardt, C. S. Ong, S. Schönauer, S. Vishwanathan, A. J. Smola, and H.-P. Kriegel, "Protein function prediction via graph kernels," *Bioinformatics*, vol. 21, no. suppl_1, pp. i47–i56, 2005.

[66] S. Pan, J. Wu, and X. Zhu, "CogBoost: Boosting for fast cost-sensitive graph classification," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 11, pp. 2933–2946, Nov. 2015.

[67] M. Welling and T. N. Kipf, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2016.

[68] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?," in *Proc. Int. Conf. Learn. Representations*, 2018.

[69] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

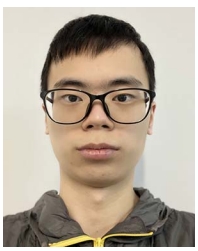
[70] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, 2018.

[71] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.
[72] F.-Y. Sun, J. Hoffmann, V. Verma, and J. Tang, "InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization," in *Proc. Int. Conf. Learn. Representations*, 2020.
[73] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6028–6039.
[74] M. Jang, S.-Y. Chung, and H. W. Chung, "Test-time adaptation via self-training with nearest neighbor information," in *Proc. Int. Conf. Learn. Representations*, 2023.
[75] N. Yin et al., "Coco: A coupled contrastive framework for unsupervised domain adaptive graph classification," 2023, *arXiv:2306.04979*.

Junyu Luo is currently working toward the PhD degree with the School of Computer Science, Peking University, Beijing, China. He has authored or coauthored several papers in top-tier venues, including *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IJCAI, and CVPR. His research interests include data-centric machine learning, with a focus on machine learning on graphs and large language models. He is particularly interested in addressing real-world challenges in graph neural networks.

Haoyu Tao is currently working toward the undergraduation degree with the School of Mathematical Sciences, Peking University, Beijing, China. His research interests include machine learning, bioinformatics, and omics data analysis.

Xiao Luo is currently an assistant professor with the Department of Statistics, University of Wisconsin–Madison, Madison, WI, USA. His research interests include machine learning, AI for science, data mining, and bioinformatics. He is an associate editor for *IEEE Transactions on Emerging Topics in Computational Intelligence*.



Yusheng Zhao is currently working toward the graduation degree wit the School of Computer Science, Peking University, Beijing, China. His research interests include machine learning with graphs, vision and language, and adversarial learning.

Zhiping Xiao received the BS degree in computer science from Peking University, Beijing, China, the MS degree from UC Berkeley, and the PhD degree from the Department of Computer Science, University of California at Los Angeles. She is currently a postdoctoral researcher with the Department of Computer Science and Engineering, University of Washington. Her current research interests include large language models, biomedical computing, social network analysis, and computing education etc.

Dailan He is currently working toward the PhD degree with the Department of Electronic Engineering. He is a member with Multimedia Laboratory, The Chinese University of Hong Kong, Hong Kong. His research interests include deep image compression, image generation, graph neural networks, and domain adaptation.

Wei Ju received the BS degree in mathematics from Sichuan University, Sichuan, China, in 2017, and the PhD degree in computer science from Peking University, Beijing, China, in 2022. He is currently a postdoc research fellow of computer science with Peking University. His research interests include in the area of machine learning on graphs including graph representation learning and graph neural networks.

Chong Chen received the BS degree in mathematics and applied mathematics and the PhD degree in statistics from Peking University in 2013 and 2019, respectively. He is currently a research scientist with Terminus Group. His research interests include statistics, machine learning, and computer vision.

Xian-Sheng Hua (Fellow, IEEE) received the BS and PhD degrees in applied mathematics from Peking University, Beijing, China, in 1996 and 2001, respectively. In 2001, he joined Microsoft Research Asia as a researcher and has been a senior researcher with Microsoft Research Redmond since 2013. He became a researcher and senior director of the Alibaba Group in 2015.

Ming Zhang (Member, IEEE) received the BS, MS, and PhD degrees in computer science from Peking University.

She is currently a full professor with the School of Computer Science, Peking University. She has authored or coauthored more than 300 research papers on Text Mining and Machine Learning in the top journals and conferences. She has won the best paper award at ICML 2014 and the best paper award at ACL 2025.