PALA: Class-imbalanced Graph Domain Adaptation via Prototype-anchored Learning and Alignment

Xin Ma[1] , **Yifan Wang**[2] , **Siyu Yi**[3] , **Wei Ju**[1*] , **Bei Wu**[4] , **Ziyue Qiao**[5] ,
Chenwei Tang[1] and **Jiancheng Lv**[1*]

[1]College of Computer Science, Sichuan University
[2]School of Information Technology & Management, University of International Business and Economics
[3]College of Mathematics, Sichuan University
[4]Pittsburgh Institute, Sichuan University
[5]School of Computing and Information Technology, Great Bay University
{maxin88, wubei}@stu.scu.edu.cn, yifanwang@uibe.edu.cn,
{siyuyi, juwei, lvjiancheng}@scu.edu.cn, ziyuejoe@gmail.com

Abstract

Graph domain adaptation is a key subfield of graph transfer learning that aims to bridge domain gaps by transferring knowledge from a label-rich source graph to an unlabeled target graph. However, most existing methods assume balanced labels in the source graph, which often fails in practice and leads to biased knowledge transfer. To address this, in this paper, we propose a prototype-anchored learning and alignment framework for class-imbalanced graph domain adaptation. Specifically, we incorporate pointwise node mutual information into the graph encoder to capture high-order topological proximity and learn generalized node representations. Leveraging this, we then introduce categorical prototypes with adversarial proto-instances for prototype-anchored learning and recalibration to represent the source graph under an imbalanced class distribution. Finally, we introduce a weighted prototype contrastive adaptation strategy that aligns target pseudo-labels with source prototypes to handle class imbalance during adaptation. Extensive experiments show that our PALA outperforms the state-of-the-art methods. Our code is available at https://github.com/maxin88scu/PALA.

1 Introduction

Graphs serve as an effective tool for representing intricate relationships across diverse domains. As a critical task in graphs [Kipf and Welling, 2017; Ju *et al.*, 2024a], node classification aims to predict the labels of individual nodes, playing a key role in areas such as community detection and molecular property prediction. Despite its importance, node classification heavily depends on labeled data, which is often scarce and costly. Graph transfer learning [Yuan *et al.*, 2023; Qiao *et al.*, 2023; Li *et al.*, 2025], which leverages learned
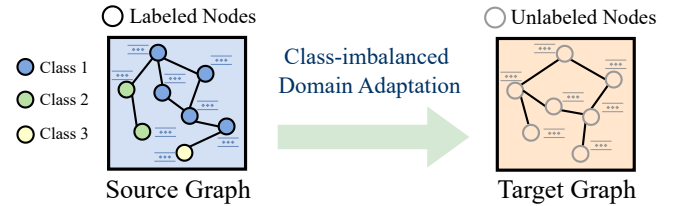
---

*Corresponding authors.

Figure 1: An illustration of the class-imbalanced domain adaptation on graphs. This task aims to enable a labeled yet class-imbalanced source graph to generalize effectively to an unlabeled target graph.

knowledge from a labeled source domain graph an unlabeled target domain graph, has become a promising solution.

In recent years, many efforts have been made for graph transfer learning. One typical type of method is the graph domain adaptation (GDA) for node classification, where the primary task is to estimate the label distribution of the target domain graph based on the labeled source graph [Cai *et al.*, 2024]. Thanks to the expressive modeling capabilities of graph neural networks (GNNs) [Ju *et al.*, 2024b; Luo *et al.*, 2023; Wang *et al.*, 2024], the recent focus has shifted toward aligning distributions of the source and target domain graphs within the representation space learned by GNNs [Luo *et al.*, 2024a; Liu *et al.*, 2024a]. Distance-based methods explicitly minimize the distance between two domains via the proposed domain discrepancy metrics [Liu *et al.*, 2024b; Wu *et al.*, 2023]. In contrast, adversarial methods use a discriminator to distinguish between domains and learn invariant features [Luo *et al.*, 2024b; Dai *et al.*, 2022].

Despite the notable performance made by these methods, they often assume a balanced label distribution in the source graph. We argue that this assumption often fails in many real-world scenarios [Ju *et al.*, 2024c], where a large portion of classes has few labeled nodes (tail classes), while a small number of classes have a disproportionately large share of labeled nodes (head classes) [Huang *et al.*, 2016; Ju *et al.*, 2025], resulting in highly imbalanced class distributions. For example, in a large-scale citation network, the

number of papers on artificial intelligence greatly exceeds those on cryptography [Li *et al.*, 2023]. As a result, the model trained on an imbalanced source graph would be severely biased and subsequently impact the graph domain adaptation process. Therefore, as shown in Figure 1, in this paper, we focus on a more practical scenario, which is referred to as *class-imbalanced graph domain adaptation* on graphs.

However, developing such an effective class-imbalanced domain adaptation framework is challenging due to the following two key obstacles: *(i) How to effectively mitigate the impact of class imbalance to extract meaningful knowledge from the source graph?* The imbalance often causes the model to focus primarily on the majority class, resulting in degraded performance on tail classes. And this ultimately leads to biased learning and poor generalization. *(ii) How to overcome domain shifts across graphs to make accurate and domain-invariant predictions on the target graph?* Existing works typically emphasize aligning the overall distributions between the source and target domain graphs, but they often overlook the fact that the class-level distributions may not be well-aligned, especially for the skewed tail classes.

To handle these challenges, our motivation is to investigate categorical prototypes, where each category has a single prototype to capture the semantic information of the source graph for adaptation. Toward this end, we propose a novel **P**rototype-**A**nchored **L**earning and **A**lignment method (term as **PALA**) for class-imbalanced graph domain adaptation, which leverages the categorical prototype instead source features to avoid the aforementioned problem. Specifically, we incorporate pointwise node mutual information into the graph encoder, allowing the model to explore high-order topological proximity and learn more generalized node representations. Based on this, we then introduce a set of categorical prototypes with corresponding adversarial proto-instances for prototype-anchored learning and recalibration to mitigate the class imbalance. Finally, we introduce a prototype-anchored contrastive adaptation strategy for class-imbalanced GDA, which generates pseudo-labels for the target domain graph and aligns them with the corresponding source prototypes to mitigate the class-imbalance issue during adaptation. To summarize, we make the following contributions:

- *New Perspective:* We focus on the class-imbalance issue in the source graph for graph domain adaptation tasks. To the best of our knowledge, this is the first work aimed at solving class-imbalanced graph domain adaptation.

- *Novel Methodology:* We propose a prototype-anchored learning and recalibration strategy to eliminate the prototype bias in the source graph and employ a novel contrastive prototype for alignment.

- *Extensive Experiments:* We conduct extensive experiments on various public datasets to evaluate our PALA. The results further show the superiority of our method in handling class-imbalanced graph domain adaptation.

2 Notations & Problem Definition

Source & Target Domain Graph. Let the *source domain graph* denoted as $\mathcal{G}^s = \{\mathcal{V}^s, \mathcal{E}^s, \boldsymbol{X}^s, \boldsymbol{Y}^s\}$, where $\mathcal{V}^s$ and $\mathcal{E}^s$ represent the labeled nodes and edge set. The node feature matrix is given by $\boldsymbol{X}^s \in \mathbb{R}^{|\mathcal{V}^s| \times F}$, where each row $x_v \in \mathbb{R}^F$ corresponds to the F-dimensional feature vector of a node v. The graph structure is captured by the adjacency matrix $\boldsymbol{A}^s \in \mathbb{R}^{|\mathcal{V}^s| \times |\mathcal{V}^s|}$, where $\boldsymbol{A}^s_{ij} = 1$ if there exists edge $(v_i, v_j) \in \mathcal{E}^s$, otherwise, $\boldsymbol{A}^s_{ij} = 0$. In the source domain graph, each node is assigned a corresponding label. The labels are imbalanced, and the label matrix is denoted as $\boldsymbol{Y}^s \in \mathbb{R}^{|\mathcal{V}^s| \times C}$, where C is the number of node classes. Similarly, we represent the *target domain graph* as $\mathcal{G}^t = \{\mathcal{V}^t, \mathcal{E}^t, \boldsymbol{X}^t\}$, where $\mathcal{V}^t$ is the set of unlabeled nodes and $\mathcal{E}^t$ is the edge set.

Class-imbalanced Graph Domain Adaptation. Given a labeled source graph $\mathcal{G}^s = \{\mathcal{V}^s, \mathcal{E}^s, X^s, Y^s\}$ and an unlabeled target graph $\mathcal{G}^t = \{\mathcal{V}^t, \mathcal{E}^t, X^t\}$, let N_c denote the number of nodes in $\mathcal{G}^s$ that belong to the c-th class, satisfying $N_1 \geq N_2 \geq \cdots \geq N_C$. The source graph is class-imbalanced and quantified by the imbalance factor (IF) defined as N_1/N_C. The goal of class-imbalanced graph domain adaptation is to mitigate the effects of this imbalance and generate accurate predictions for the unlabeled nodes in $\mathcal{G}^t$.

3 Proposed Method

As shown in Figure 2, our proposed PALA consists of three modules: (1) *Graph Consistency Encoder*, designed to capture high-order structural information of the graph for learning more generalized node representations; (2) *Prototype-Anchored Learning and Recalibration*, which generates categorical prototypes by reducing contrastive negative pair redundancy, mining distinguishable positive pairs, and recalibrating prototypes to eliminate bias; (3) *Contrastive Prototype Adaptation*, which aligns pseudo-labeled target features with source prototypes to facilitate graph domain adaptation.

3.1 Graph Consistency Encoder

To effectively capture the high-order structural dependencies in the graph, we consider the global consistency relationships between nodes and calculate positive pointwise mutual information (PPMI) [Zhuang and Ma, 2018] to learn an informative representation of the node. Specifically, for a given graph $\mathcal{G}^*, * \in \{s, t\}$, we employ random walk on $\boldsymbol{A}^*$ to sample a set of paths and calculate the frequency matrix $\boldsymbol{F}^*$ between nodes to count their co-occurrence. Then, the PPMI between nodes can be computed as:

$$\mathbb{P}^*_{ij} = \frac{\boldsymbol{F}^*_{ij}}{\sum_{i,j} \boldsymbol{F}^*_{ij}}, \mathbb{P}^*_i = \frac{\sum_j \boldsymbol{F}^*_{ij}}{\sum_{i,j} \boldsymbol{F}^*_{ij}}, \mathbb{P}^*_j = \frac{\sum_i \boldsymbol{F}^*_{ij}}{\sum_{i,j} \boldsymbol{F}^*_{ij}},$$
$$\boldsymbol{P}^*_{ij} = \max\{\log(\frac{\mathbb{P}^*_{ij}}{\mathbb{P}^*_i \times \mathbb{P}^*_j}), 0\}, \tag{1}$$

where $\mathbb{P}_{ij}$ represent the probabilities that v_j in the v_i's context with a predefined window. $\boldsymbol{P}^*_{ij}$ captures the high-order topological proximity between nodes. A higher value of $\boldsymbol{P}^*_{ij}$ indicates that the two nodes have a higher co-occurrence frequency compared to when they are independent. Then, we can reformulate $\boldsymbol{P}^*$ as the new adjacency matrix and the global consistency knowledge can be extracted as:

$$\boldsymbol{Z}^{(l)}_* = \sigma(\boldsymbol{D}^{*-\frac{1}{2}} \tilde{\boldsymbol{P}}^* \boldsymbol{D}^{*-\frac{1}{2}} \boldsymbol{Z}^{(l-1)}_* \boldsymbol{W}^{*(l)}), \tag{2}$$
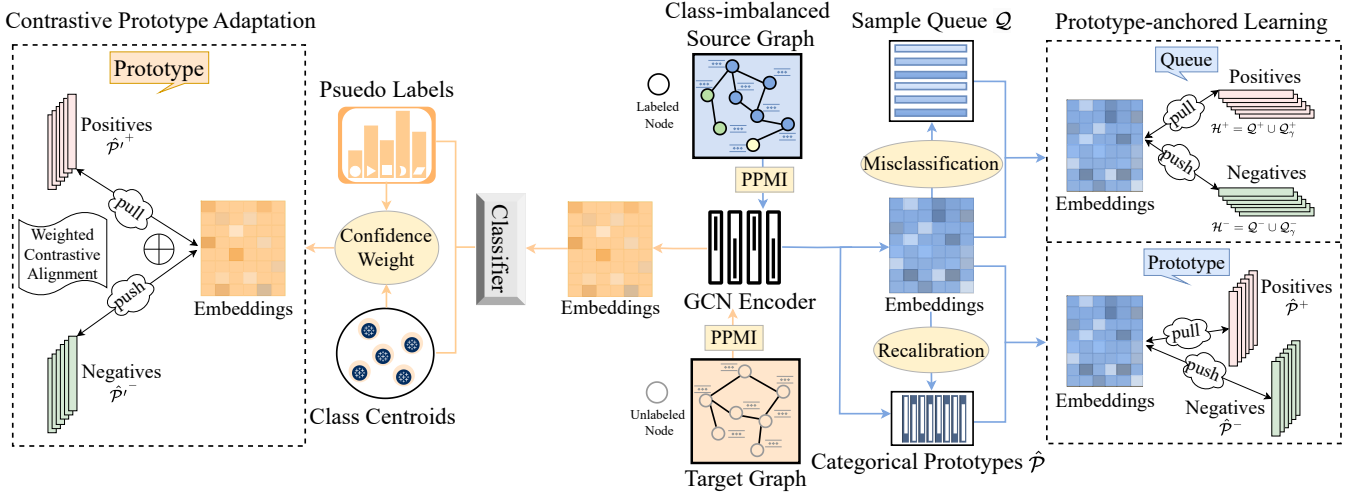
Figure 2: Illustration of the proposed framework PALA, which contains three modules: (1) **Graph Consistency Encoder**: GCN encoder with $\boldsymbol{P}^*, * \in \{s, t\}$ to capture generalized information of $\mathcal{G}^s$ and $\mathcal{G}^t$. (2) **Prototype-anchored Recalibration & Learning**: With categorical prototypes, we maintain a sample queue, construct adversarial proto-instances, and recalibrate the prototype to eliminate bias. (3) **Contrastive Prototype Adaptation**: We generate pseudo-labels for the target domain and employ weighted contrastive alignment for adaptation.

where $\tilde{\boldsymbol{P}}^* = \tilde{\boldsymbol{P}} + \boldsymbol{I}$, $\boldsymbol{D}_i^* = \sum_j \tilde{\boldsymbol{P}}_{ij}^*$ and $\boldsymbol{W}^{*(l)}$ is the filter weight of l-th layer. By stacking L layers, the global consistency knowledge can be extracted as $\boldsymbol{Z}^* = \boldsymbol{Z}_*^{(L)} \in \mathbb{R}^d$.

3.2 Prototype-anchored Recalibration & Learning

Since the categorical prototypes have a generalization ability for class-imbalanced data where each class corresponds to only one prototype, we leverage the categorical prototype to extract unbiased knowledge from the source domain graph.

Prototype-anchored Construction. To generate the prototype, we randomly initialize a set of parameters $\mathcal{P} = \{\boldsymbol{p}_c \in \mathbb{R}^d | 1 \leq c \leq C\}$ and learn them in a contrastive manner. Given that head classes disproportionately dominate negative pairs in conventional contrastive methods, hindering tail class learning, we address this by reducing redundant negative pairs and identifying meaningful positive pairs. In practice, we maintain an instance queue that contains misclassified instance $\mathcal{Q}$ for prototype learning. For each node v_i in $\mathcal{G}^s$, $\mathcal{Q}$ is divided into $\mathcal{Q}^+ = \{v_j | v_j \in \mathcal{Q}, y_j = y_i\}$ and $\mathcal{Q}^- = \{v_j | v_j \in \mathcal{Q}, y_j \neq y_i\}$. Similarly, we also divide prototype set into $\mathcal{P}^+ = \{\boldsymbol{p}_c^+ | c = y_i\}$ and $\mathcal{P}^- = \{\boldsymbol{p}_c^- | c \neq y_i\}$. Then, we synthesize the adversarial proto-instances to rectify the decision boundaries of tail classes during prototype learning. For the negative proto-instances, we rank the distance between node v_i and nodes in $\mathcal{Q}^-$ in ascending order and construct the adversarial negative proto-instance set $\mathcal{Q}_\gamma^-$:

$$\mathcal{Q}_\gamma^- = \{v_j | v_j \in \mathcal{Q}^-, d(\boldsymbol{z}_j^s, \boldsymbol{z}_i^s) \leq d(\boldsymbol{z}_j^s, \boldsymbol{z}_\gamma^s)\}, \quad (3)$$

where $d(\cdot, \cdot)$ denote the distance between nodes based on their representation in $\boldsymbol{Z}^s$, $\boldsymbol{z}_\gamma^s$ is embedding of the γ-th node in $\mathcal{Q}^-$. For harder negative instances under imbalance class distribution, we randomly perturb each element in $\mathcal{Q}_\gamma^-$ with the

positive prototype $\boldsymbol{p}_c^+$, defined as:

$$\hat{\mathcal{Q}}_\gamma^- = \left\{ \frac{(1 - \epsilon_j)\boldsymbol{z}_j^s + \epsilon_j \boldsymbol{p}_c^+}{\|(1 - \epsilon_j)\boldsymbol{z}_j^s + \epsilon_j \boldsymbol{p}_c^+\|_2} \Big| v_j \in \mathcal{Q}_\gamma^- \right\}, \quad (4)$$

where ϵ_i denotes the interpolation coefficient. Similarly, for the positive proto-instances, we select all the samples in $\mathcal{Q}^+$ and combine them with incorrectly assigned prototype $\boldsymbol{p}_c^-$.

Prototype Recalibration. To further alleviate prototype bias during the learning process, where proto-instances tend to favor head classes and undermine performance on tail classes, we propose a prototype recalibration strategy, evaluating the importance of each categorical prototype and emphasizing the contribution of the tail class. Given the similarity between the node feature and prototype, we introduce a rectified sigmoid function for prototype recalibration. The calibration factor for each prototype can be defined as:

$$\omega_c = \frac{1}{N_c} \sum_{1 \leq i \leq N_c} \frac{1}{1 + \exp(-(\boldsymbol{z}_i^s)^\top \cdot \boldsymbol{p}_c)}. \quad (5)$$

To ensure stability, we progressively updated the calibration factor via the exponential moving average (EMA):

$$\bar{\omega}_c = \beta \cdot \bar{\omega}_c + (1 - \beta) \cdot \omega_c, \quad (6)$$

where β denotes the smoothing coefficient. A larger recalibration factor indicates a higher representativeness of the prototype in guiding the model's learning process. Eventually, we impose all prototypes by the recalibration factor:

$$\hat{\mathcal{P}} = \{\log(\bar{\omega}_c) + \boldsymbol{p}_c | \boldsymbol{p}_c \in \mathcal{P}\}. \quad (7)$$

Prototype-anchored Learning. Inspired by the success of supervised contrastive learning (SCL) in delivering stable performance [Graf *et al.*, 2021]. To address class imbalance more effectively, we extend SCL by incorporating prototypes

and both positive and negative adversarial proto-instances into the training process. The prototype-anchored learning loss in the source domain graph can be formulated as:

$$
\begin{aligned}
\mathcal{L}_p = \sum_{i=1}^{|\mathcal{V}_s|} \log\Big[1 + \Big(\sum_{v_j \in \mathcal{H}^-} \exp((\boldsymbol{z}_i^s)^\top \cdot \boldsymbol{z}_j) + \sum_{\boldsymbol{p}_c \in \hat{\mathcal{P}}^-} \exp((\boldsymbol{z}_j^s)^\top \cdot \boldsymbol{p}_c)\Big) \\
\cdot \Big(\sum_{v_j \in \mathcal{H}^+} \exp(-(\boldsymbol{z}_i^s)^\top \cdot \boldsymbol{z}_j) + \sum_{\boldsymbol{p}_c \in \hat{\mathcal{P}}^+} \exp(-(\boldsymbol{z}_j^s)^\top \cdot \boldsymbol{p}_c)\Big)\Big],
\end{aligned}
\tag{8}
$$

where $\mathcal{H}^* = \mathcal{Q}^* \cup \hat{\mathcal{Q}}_\gamma^*, * \in \{+,-\}$.

3.3 Contrastive Prototype Adaptation

Given the prototype set of the source domain graph, directly aligning to the target domain graph is nontrivial due to the lack of target annotations. In this part, we generate pseudo-labels for the target domain and introduce the weighted contrastive prototype alignment for adaptation.

Target Pseudo-Labeling. Given the unlabeled target domain graph $\mathcal{G}^t$, we aim to conduct domain alignment based on the generated prototype in source domain graph $\mathcal{G}^s$. Since the target node annotations are unavailable, we apply a self-supervised labeling strategy to generate the target pseudo labels. Specifically, based on the encoded target domain graph embedding $\boldsymbol{Z}_t$, we predict the probability $\hat{y}_i^k = g(\boldsymbol{z}_i^t)$ for each class k with an MLP-based classifier $g(\cdot)$. Then, we attain the centroid for each class in $\mathcal{G}^t$ as:

$$
\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{|\mathcal{V}_t|} \hat{y}_i^k \boldsymbol{q}_i}{\sum_{i=1}^{|\mathcal{V}_t|} \hat{y}_i^k}.
\tag{9}
$$

These centroids provide a robust and reliable representation of the category distributions within the target domain [Liang et al., 2020; Qiu et al., 2021], and we obtain the pseudo label via the nearest centroid as:

$$
\bar{y}_i = \arg_k \max \phi(\boldsymbol{z}_i^t, \boldsymbol{\mu}_k),
\tag{10}
$$

where $\phi(\cdot, \cdot)$ denote the cosine similarity. Note that we can compute the new target centroid based on the new pseudo labels for the iteration process.

Weighted Contrastive Alignment. Based on the generated pseudo labels of the target domain graph, we align them to the source prototype $\hat{\mathcal{P}}$. Since the target pseudo labels may be noisy, we assume more reliable samples are generally closer to the class centroid [Chen et al., 2019] and employ a weighted contrastive alignment for adaptation. The confidence of node v_i in $\mathcal{G}^t$ can be compute as:

$$
r_i = \frac{\exp(\phi(\boldsymbol{z}_i^t, \boldsymbol{\mu}_{\bar{y}_i})/\tau)}{\sum_{k=1}^K \exp(\phi(\boldsymbol{z}_i^t, \boldsymbol{\mu}_k)/\tau)},
\tag{11}
$$

where τ denotes the temperature parameter. We take each node v_i in $\mathcal{G}^t$ as an anchor, and the prototype adaptation can be conducted via the weighted contrastive alignment:

$$
\mathcal{L}_{align} = \sum_{v_i \in \mathcal{V}_t} -r_i \log\left(\frac{\exp((\boldsymbol{z}_i^t)^\mathrm{T} \cdot \boldsymbol{p}_i^+/\tau)}{\sum_{c=1}^C \exp((\boldsymbol{z}_i^t)^\mathrm{T} \cdot \boldsymbol{p}_c/\tau)}\right),
\tag{12}
$$

Algorithm 1 The overall training process of PALA

Input: Labeled source graph $\mathcal{G}^s = \{\mathcal{V}^s, \mathcal{E}^s, \boldsymbol{X}^s, \boldsymbol{Y}^s\}$ with C imbalanced classes, unlabeled target graph $\mathcal{G}^t = \{\mathcal{V}^t, \mathcal{E}^t, \boldsymbol{X}^t\}$.
Output: Parameters of graph consistency encoder, prototype set.

1: Consider global consistency relationships and calculate PPMI between nodes $\boldsymbol{P}^*, *\{s,t\}$ for $\mathcal{G}^s$ and $\mathcal{G}^t$;   // Eq. 1
2: **while** not convergence **do**
3:      Encode high-order structure information of the graph based on $\boldsymbol{P}^*$ to get $Z^*, * \in \{s,t\}$;   // Eq. 2
     /* Prototype Recalibration & Learning */
4:      Construct adversarial proto-instances $\hat{\mathcal{Q}}_\gamma^*, * \in \{+,-\}$ to rectify the boundaries of tail classes in $\mathcal{G}^s$; // Eqs. 3-4
5:      Initialize categorical prototypes $\mathcal{P}$ and recalibrate them to get $\hat{\mathcal{P}}$ to alleviate prototype bias ;   // Eqs. 5-7
6:      Prototype-anchored learning with $\hat{\mathcal{Q}}_\gamma^*$ and $\hat{\mathcal{P}}$;   // Eq. 8
     /* Contrastive Prototype Adaptation */
7:      Attain class centroid $\{\boldsymbol{\mu}_k\}, k \in \{1, \ldots, C\}$ and pseudo labels for the target graph $\mathcal{G}^t$;   // Eq. 10
8:      Weighted prototype contrastive alignment based on $\hat{\mathcal{P}}$ from source and target pseudo labels ;   // Eqs. 11-12
9:      Early learning regularization for class-imbalanced contrastive alignment process ;   // Eqs. 13-14
10: **end while**
11: Obtain graph consistency encoder, prototype set for adaptation.

---

where $\boldsymbol{p}_i^+$ denotes the prototype with the same class to v_i, where the label are attained from target pseudo-labeling.

Adaptation Regularization. Since learning from class-imbalanced data always first memorizes the unbiased samples and then the biased knowledge [Arpit et al., 2017], we further regularize the adaptation process by leveraging the early predictions of each sample. Formally, we maintain a memory bank $\mathcal{H} = \{\bar{\boldsymbol{h}}_1, \ldots, \bar{\boldsymbol{h}}_{\mathcal{V}_t}\}$ to record the prediction of node v_i and update the bank via EMA:

$$
\bar{\boldsymbol{h}}_i = \beta\bar{\boldsymbol{h}}_i + (1 - \beta)\boldsymbol{h}_i,
\tag{13}
$$

where $\boldsymbol{h}_{i,k}$ is the alignment probability of C prototypes computed as in Eq. 12. We further regularize the model via an early learning loss [Liu et al., 2020]:

$$
\mathcal{L}_{elr} = \log(1 - (\boldsymbol{h}_i)^\mathrm{T}\bar{\boldsymbol{h}}_i).
\tag{14}
$$

This regularization ensures that the current prediction aligns closely with the average prediction.

3.4 Overall Optimization

For the class-imbalance of nodes in the source domain graph, we apply the logit compensation to eliminate the bias and learn the rectification of the consistency knowledge [Menon et al., 2021], denoted as:

$$
\mathcal{L}_{lc} = -\lambda_c \log \frac{\exp(\varphi_y(\boldsymbol{z}_i^s) + \delta_c)}{\sum_{c'=1}^C \exp(\varphi_{c'}(\boldsymbol{z}_i^s) + \delta_{c'})},
\tag{15}
$$

where $\varphi(\cdot)$ represents the classification function output the logit of labels, λ_y is the importance weight of class y_i^s and δ_c is the compensation value of class c. In our paper, we set $\lambda_c = 1$ and $\delta_c = \log N_c$ as in the work [Menon et al., 2021]. Finally, the overall objective of our framework is defined as:

$$
\mathcal{L} = \mathcal{L}_{lc} + \mathcal{L}_p + \mathcal{L}_{align} + \mathcal{L}_{elr}.
\tag{16}
$$

The detailed algorithm framework is provided in Algorithm 1.

Methods	IF (ρ)	A⇒C Micro	A⇒C Macro	A⇒D Micro	A⇒D Macro	C⇒A Micro	C⇒A Macro	C⇒D Micro	C⇒D Macro	D⇒A Micro	D⇒A Macro	D⇒C Micro	D⇒C Macro
GCN	10	$66.01_{\pm1.15}$	$60.90_{\pm2.14}$	$61.21_{\pm0.57}$	$53.20_{\pm0.71}$	$58.25_{\pm1.22}$	$56.52_{\pm2.87}$	$63.39_{\pm0.32}$	$58.38_{\pm1.27}$	$55.47_{\pm0.39}$	$52.80_{\pm0.86}$	$61.68_{\pm0.38}$	$58.35_{\pm0.68}$
	20	$62.28_{\pm1.17}$	$55.48_{\pm1.79}$	$60.75_{\pm0.69}$	$51.60_{\pm1.48}$	$51.17_{\pm1.58}$	$43.78_{\pm1.54}$	$59.00_{\pm0.88}$	$48.22_{\pm1.57}$	$50.93_{\pm2.70}$	$45.12_{\pm4.44}$	$54.02_{\pm0.27}$	$46.18_{\pm0.55}$
	50	$53.07_{\pm3.39}$	$42.41_{\pm3.32}$	$56.26_{\pm2.21}$	$43.35_{\pm3.53}$	$44.38_{\pm1.12}$	$32.20_{\pm2.51}$	$53.06_{\pm0.55}$	$37.09_{\pm1.18}$	$41.86_{\pm1.31}$	$31.11_{\pm1.50}$	$42.28_{\pm3.28}$	$32.05_{\pm3.30}$
GAT	10	$66.48_{\pm0.72}$	$58.37_{\pm2.74}$	$61.58_{\pm0.75}$	$51.44_{\pm1.02}$	$55.10_{\pm2.47}$	$51.54_{\pm4.12}$	$60.87_{\pm1.00}$	$52.99_{\pm1.34}$	$56.18_{\pm0.31}$	$53.11_{\pm0.80}$	$61.64_{\pm0.23}$	$58.22_{\pm0.53}$
	20	$61.63_{\pm3.34}$	$50.54_{\pm3.72}$	$61.04_{\pm1.87}$	$50.10_{\pm1.46}$	$47.18_{\pm1.04}$	$37.30_{\pm1.61}$	$55.95_{\pm1.26}$	$43.32_{\pm2.17}$	$49.18_{\pm1.59}$	$42.20_{\pm2.78}$	$53.05_{\pm3.25}$	$54.29_{\pm2.63}$
	50	$53.21_{\pm7.62}$	$42.07_{\pm7.64}$	$53.67_{\pm4.00}$	$40.36_{\pm5.33}$	$41.02_{\pm0.99}$	$26.41_{\pm1.18}$	$49.76_{\pm0.63}$	$32.29_{\pm1.15}$	$40.14_{\pm1.46}$	$28.89_{\pm1.85}$	$45.38_{\pm7.90}$	$34.77_{\pm8.31}$
GIN	10	$62.12_{\pm0.69}$	$55.03_{\pm0.84}$	$60.91_{\pm0.79}$	$54.39_{\pm1.48}$	$54.35_{\pm1.75}$	$49.21_{\pm1.31}$	$60.14_{\pm1.00}$	$54.51_{\pm2.16}$	$52.59_{\pm0.67}$	$49.43_{\pm2.42}$	$59.89_{\pm1.42}$	$54.68_{\pm2.28}$
	20	$60.03_{\pm1.03}$	$50.98_{\pm0.94}$	$59.12_{\pm1.05}$	$48.27_{\pm0.92}$	$50.38_{\pm2.75}$	$42.80_{\pm3.88}$	$55.87_{\pm2.47}$	$46.42_{\pm4.71}$	$47.79_{\pm1.49}$	$41.02_{\pm2.77}$	$53.47_{\pm2.49}$	$44.83_{\pm2.61}$
	50	$54.39_{\pm1.62}$	$44.28_{\pm1.69}$	$54.43_{\pm1.60}$	$41.90_{\pm2.09}$	$41.26_{\pm0.37}$	$29.45_{\pm0.51}$	$47.05_{\pm0.60}$	$31.57_{\pm1.13}$	$43.38_{\pm1.26}$	$34.41_{\pm2.05}$	$48.59_{\pm2.15}$	$39.22_{\pm2.29}$
GDA-SpecReg	10	$51.27_{\pm1.86}$	$39.29_{\pm2.43}$	$53.69_{\pm3.02}$	$40.87_{\pm5.56}$	$49.69_{\pm4.93}$	$39.08_{\pm7.51}$	$57.41_{\pm3.15}$	$45.54_{\pm4.35}$	$52.73_{\pm3.20}$	$47.07_{\pm7.86}$	$56.37_{\pm1.40}$	$45.04_{\pm3.46}$
	20	$49.68_{\pm1.42}$	$36.44_{\pm1.70}$	$51.34_{\pm3.70}$	$35.82_{\pm3.00}$	$48.11_{\pm4.07}$	$38.20_{\pm6.83}$	$51.64_{\pm1.96}$	$35.15_{\pm3.49}$	$43.84_{\pm3.79}$	$31.89_{\pm3.72}$	$44.53_{\pm6.48}$	$31.21_{\pm6.80}$
	50	$49.80_{\pm6.48}$	$36.09_{\pm7.63}$	$47.44_{\pm4.58}$	$29.36_{\pm6.49}$	$47.64_{\pm4.97}$	$33.87_{\pm7.90}$	$52.63_{\pm2.53}$	$38.46_{\pm6.93}$	$41.95_{\pm3.49}$	$28.28_{\pm7.97}$	$45.63_{\pm5.22}$	$30.81_{\pm8.85}$
A2GNN	10	$61.23_{\pm0.93}$	$54.72_{\pm2.23}$	$59.00_{\pm0.64}$	$46.61_{\pm1.16}$	$51.29_{\pm0.51}$	$41.70_{\pm0.77}$	$61.07_{\pm0.32}$	$50.09_{\pm0.68}$	$59.54_{\pm0.83}$	$55.41_{\pm1.74}$	$60.32_{\pm1.82}$	$60.11_{\pm2.30}$
	20	$35.95_{\pm0.99}$	$25.28_{\pm1.45}$	$49.06_{\pm1.28}$	$35.11_{\pm1.11}$	$46.09_{\pm0.19}$	$33.40_{\pm0.39}$	$56.91_{\pm0.24}$	$42.06_{\pm0.38}$	$41.31_{\pm0.79}$	$31.44_{\pm1.20}$	$41.00_{\pm0.70}$	$31.44_{\pm1.04}$
	50	$32.48_{\pm1.62}$	$17.61_{\pm1.15}$	$35.22_{\pm0.12}$	$15.10_{\pm0.24}$	$40.26_{\pm0.08}$	$24.09_{\pm0.07}$	$50.33_{\pm0.17}$	$31.27_{\pm0.22}$		$14.29_{\pm0.48}$	$27.79_{\pm0.35}$	$13.19_{\pm0.72}$
GraphENS	10	$70.49_{\pm0.63}$	$64.53_{\pm1.33}$	$66.43_{\pm1.40}$	$58.40_{\pm3.69}$	$63.25_{\pm1.11}$	$62.41_{\pm2.29}$	$67.47_{\pm0.37}$	$63.87_{\pm1.23}$	$58.95_{\pm0.64}$	$58.34_{\pm1.07}$	$65.50_{\pm0.53}$	$63.52_{\pm0.65}$
	20	$68.12_{\pm1.26}$	$57.59_{\pm1.13}$	$64.41_{\pm0.70}$	$53.09_{\pm1.11}$	$59.61_{\pm1.30}$	$51.96_{\pm1.31}$	$65.58_{\pm1.02}$	$57.31_{\pm3.39}$	$54.72_{\pm0.46}$	$49.54_{\pm1.89}$	$61.11_{\pm0.49}$	$52.47_{\pm1.99}$
	50	$62.74_{\pm0.81}$	$51.76_{\pm0.79}$	$61.01_{\pm1.66}$	$47.74_{\pm4.14}$	$54.09_{\pm1.51}$	$45.51_{\pm1.74}$	$60.40_{\pm0.90}$	$47.81_{\pm2.29}$	$50.67_{\pm0.43}$	$42.18_{\pm0.44}$	$56.26_{\pm0.85}$	$47.81_{\pm2.43}$
TAM	10	$72.36_{\pm0.50}$	$67.27_{\pm0.56}$	$67.12_{\pm0.69}$	$61.52_{\pm1.05}$	$63.40_{\pm0.41}$	$63.40_{\pm1.34}$	$70.10_{\pm0.85}$	$66.20_{\pm1.02}$	$59.17_{\pm0.68}$	$55.36_{\pm1.29}$	$66.65_{\pm0.39}$	$63.16_{\pm0.77}$
	20	$68.90_{\pm1.49}$	$60.01_{\pm1.23}$	$66.11_{\pm0.99}$	$56.88_{\pm1.83}$	$61.36_{\pm0.70}$	$57.64_{\pm2.19}$	$68.11_{\pm0.69}$	$\mathbf{62.57_{\pm1.98}}$	$53.81_{\pm1.05}$	$46.92_{\pm1.65}$	$61.23_{\pm1.22}$	$53.28_{\pm1.42}$
	50	$64.93_{\pm3.74}$	$55.77_{\pm4.74}$	$61.63_{\pm0.99}$	$50.23_{\pm1.48}$	$58.95_{\pm1.64}$	$\mathbf{50.99_{\pm2.06}}$	$64.66_{\pm1.19}$	$54.71_{\pm2.03}$	$53.39_{\pm1.61}$	$44.93_{\pm2.14}$	$57.88_{\pm1.67}$	$47.52_{\pm2.02}$
GraphSHA	10	$73.20_{\pm0.73}$	$\mathbf{70.43_{\pm1.14}}$	$61.79_{\pm0.09}$	$56.09_{\pm1.47}$	$66.07_{\pm1.26}$	$66.24_{\pm1.34}$	$63.05_{\pm0.93}$	$56.69_{\pm4.64}$	$63.28_{\pm0.55}$	$64.06_{\pm0.54}$	$71.11_{\pm0.64}$	$69.03_{\pm0.64}$
	20	$71.00_{\pm1.58}$	$64.33_{\pm2.56}$	$57.95_{\pm2.55}$	$47.37_{\pm3.16}$	$62.87_{\pm1.74}$	$\mathbf{60.41_{\pm2.91}}$	$62.72_{\pm3.73}$	$53.91_{\pm4.00}$	$57.56_{\pm1.71}$	$53.22_{\pm4.22}$	$66.93_{\pm1.40}$	$62.03_{\pm1.82}$
	50	$64.97_{\pm2.30}$	$53.99_{\pm2.49}$	$65.96_{\pm1.44}$	$54.07_{\pm1.91}$	$59.19_{\pm1.58}$	$49.30_{\pm1.53}$	$56.10_{\pm5.25}$	$44.72_{\pm6.29}$	$50.08_{\pm0.95}$	$41.72_{\pm3.42}$	$57.89_{\pm1.59}$	$50.52_{\pm2.26}$
PALA (Ours)	10	$\mathbf{77.40_{\pm0.78}}$	$67.73_{\pm2.30}$	$\mathbf{73.59_{\pm0.29}}$	$\mathbf{66.70_{\pm2.57}}$	$\mathbf{69.76_{\pm0.32}}$	$\mathbf{70.30_{\pm0.73}}$	$\mathbf{73.50_{\pm1.17}}$	$\mathbf{66.67_{\pm0.66}}$	$\mathbf{67.69_{\pm0.94}}$	$\mathbf{67.80_{\pm1.33}}$	$\mathbf{77.09_{\pm0.92}}$	$\mathbf{74.40_{\pm2.08}}$
	20	$\mathbf{76.18_{\pm1.28}}$	$\mathbf{66.91_{\pm2.90}}$	$\mathbf{72.13_{\pm1.94}}$	$\mathbf{60.59_{\pm3.02}}$	$\mathbf{64.45_{\pm2.85}}$	$53.68_{\pm2.85}$	$\mathbf{71.30_{\pm1.27}}$	$58.38_{\pm1.84}$	$\mathbf{63.73_{\pm1.50}}$	$\mathbf{62.90_{\pm1.61}}$	$\mathbf{70.12_{\pm2.45}}$	$\mathbf{66.26_{\pm2.86}}$
	50	$\mathbf{76.82_{\pm0.26}}$	$\mathbf{63.79_{\pm0.32}}$	$\mathbf{72.63_{\pm0.79}}$	$\mathbf{59.45_{\pm0.71}}$	$\mathbf{61.58_{\pm2.11}}$	$50.66_{\pm2.76}$	$\mathbf{66.42_{\pm2.50}}$	$\mathbf{56.14_{\pm1.28}}$	$\mathbf{59.79_{\pm1.65}}$	$\mathbf{53.00_{\pm2.89}}$	$\mathbf{67.74_{\pm0.76}}$	$\mathbf{58.22_{\pm0.32}}$

Table 1: The domain adaptation performance of all methods under different imbalance factors (ρ). A⇒C indicates that A is the source graph and C is the target graph. Micro and Macro represent micro average score(%) and macro average score(%), respectively.

4 Experiments

4.1 Experimental Settings

Datasets. We conduct extensive experiments on three real-world graph datasets: ACMv9 (A), Citationv1 (C) and DBLPv7 (D) from ArnetMiner [Tang *et al.*, 2008], constructed from distinct datasets over different periods: ACM (post-2010), Microsoft Academic Graph (pre-2008), and DBLP (2004-2008), leading to varied domain distributions. In each graph, nodes represent papers, with edges indicating undirected citation relationships. In order to simulate the real-world imbalance scenario more effectively, we adopt the same approach as [Park *et al.*, 2021] to handle the source domain data. Specifically, we iteratively process the classes with fewer nodes in the source domain to meet the imbalance factor (IF), defined as $\rho = N_1/N_C$, where $N_1 \geq N_2 \geq \cdots \geq N_C$, and N_c represents the number of nodes belonging to the c-th class in the source graph.

Baselines. For our class-imbalanced graph domain adaptation, we select three classical GNN models: GCN [Kipf and Welling, 2017], GAT [Veličković *et al.*, 2018] and GIN [Xu *et al.*, 2018] as basic baselines. Additionally, we choose two models specifically designed for domain adaptation tasks: GDA-SpecReg [You *et al.*, 2023] and A2GNN [Liu *et al.*, 2024a], and three models known for handling imbalanced data effectively: GraphENS [Park *et al.*, 2021], TAM [Song *et al.*, 2022] and GraphSHA [Li *et al.*, 2023].

Implementation Details. For the datasets mentioned above, we sequentially select one as the source domain and the other two as target domains for experiments. The imbalance factor ρ for the source domain is set to 10/20/50, covering scenarios from slightly imbalanced to severely imbalanced, to simulate

real-world data imbalance scenarios. For our PALA, we use GCN [Kipf and Welling, 2017] as the backbone encoder, with the feature dimension of 512. We set our hyperparameters γ as 5, β as 0.9, and ϵ as 0.2, which are discussed in detail in Section 4.4, and the number of training epochs is set to 200. For a fair comparison, we use the same encoder to the baselines and tuned them to optimal performance. We evaluate model performance based on two F1 scores on the target domain: *macro average* and *micro average*. We record the average of five training runs for performance evaluation.

4.2 Performance Comparison

Table 1 presents the performance of different approaches on three datasets. In the domain adaptation across all six data pairs, our PALA highlights the best-performing results. As the imbalance factor increases—i.e., the head-class nodes grow and the tail-class nodes decrease—the performance of all methods declines. This is expected, as the abundance of head-class nodes leads the model to focus more on them, neglecting the tail-class nodes. Classic GNNs lack the capability to handle domain adaptation and imbalanced data, resulting in subpar performance, especially with high imbalance facotrs. Interestingly, some baseline models designed for domain adaptation perform worse than classical GNNs. In contrast, models designed to handle imbalanced data generally perform better. This suggests that addressing data imbalance in the source domain outweighs domain discrepancy. Our method PALA performs the best in domain adaptation across most data pairs, maintaining robust performance as imbalance factors increase. When using DBLPv7 as the source domain, its smaller number of nodes typically results in worse performance compared to ACMv9 or Citationv1. However, our method effectively alleviates this issue, enhancing robust-
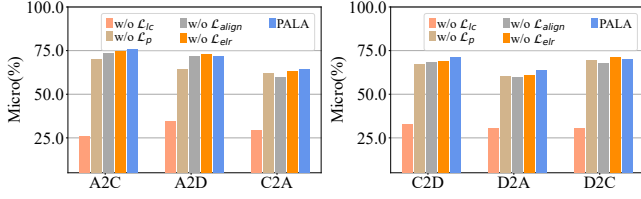
Figure 3: The results of ablation study ($\rho = 20$, Micro score (%)).
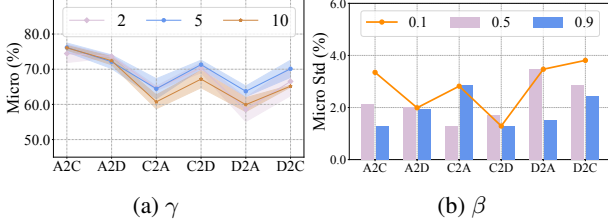
Figure 4: The results of sensitivity analysis of γ and β. Micro Std in (b) represents the standard deviation of the micro average scores.

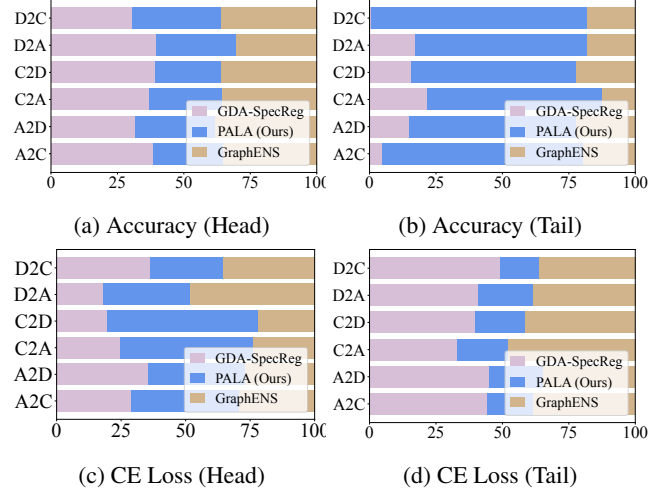| ϵ	A⇒C		A⇒D		C⇒A	
	Micro	Macro	Micro	Macro	Micro	Macro
0.2	**76.18**$_{\pm 1.28}$	**66.91**$_{\pm 2.90}$	**72.13**$_{\pm 1.94}$	60.59$_{\pm 3.02}$	**64.45**$_{\pm 2.85}$	**53.68**$_{\pm 2.85}$
0.4	75.20$_{\pm 1.80}$	63.16$_{\pm 2.51}$	72.02$_{\pm 1.07}$	**60.91**$_{\pm 1.74}$	62.76$_{\pm 3.27}$	51.86$_{\pm 3.33}$
0.8	73.32$_{\pm 2.01}$	61.93$_{\pm 2.22}$	71.94$_{\pm 2.12}$	59.71$_{\pm 0.90}$	63.98$_{\pm 0.80}$	53.56$_{\pm 0.78}$

Table 2: The results of sensitivity analysis of ϵ.

Figure 5: Accuracy and cross-entropy (CE) loss for the head and tail classes. The horizontal axis represents the normalized value for each method, i.e., the percentage of each method's value relative to the total values of the three methods.

4.3 Ablation Study

In this subsection, we design an ablation study to verify the contribution of each component in our method. We sequentially remove four losses while keeping the other losses unchanged and conduct experiments in an environment with $\rho = 20$. The results are shown in Figure 3. It can be observed that removing the logit compensation loss (w/o $\mathcal{L}_{lc}$) significantly drops performance, as the model loses valuable label information from the source domain. Similarly, removing the prototype-anchored learning loss (w/o $\mathcal{L}_p$) can address data imbalance and promote generalization, but it leads to a significant performance drop and worsens as the imbalance factor increases. Additionally, removing the weighted contrastive alignment loss (w/o $\mathcal{L}_{align}$), which aligns the feature spaces between the source and target domains, causes some degradation, but the impact is smaller than removing the source domain contrastive learning loss. This indicates that data imbalance in the source domain has a greater effect than domain shift. Finally, removing the early learning loss (w/o $\mathcal{L}_{elr}$) results in a smaller performance drop compared to other losses, but increases the variance in the domain adaptation process across most data pairs. This suggests that the early learning loss helps mitigate the noise impact and enhances robustness.

4.4 Sensitivity Analysis

Here, we conduct a sensitivity analysis on three key hyperparameters at $\rho = 20$ to verify their impact on performance.

Effect of γ. We test γ in $\{2, 5, 10\}$. The results are shown in Figure 4a. It is observed that both too small and too large γ values harm performance—small values of γ limit the positive and negative samples, reducing the model's ability to capture key features, while larger values of γ amplify data imbalance by favoring head-class samples over tail-class samples. Also, larger γ values also increase computational cost.

Effect of β. We select β from $\{0.1, 0.5, 0.9\}$ and conduct experiments. The results are shown in Figure 4b. As β increases, the model's performance becomes more stable, and the standard deviation decreases accordingly. β controls the update speed of the memory bank when calculating $\mathcal{L}_{elr}$. As β increases, the memory bank updates more slowly, encouraging the model to achieve greater stability in performance.

Effect of ϵ. We test ϵ values of $\{0.2, 0.4, 0.8\}$ and Table 2 shows the impact of different ϵ values. As ϵ increases, the model's performance declines because ϵ affects the linear interpolation of positive and negative samples in contrastive learning on the source domain. As ϵ increases, the prototype's weight increases, while the queue $\mathcal{Q}$ weight decreases, since $\mathcal{Q}$ is composed of misclassified samples, which are more likely to lead to errors. Prioritizing $\mathcal{Q}$ can improve the model more quickly. When ϵ is large, the model ignores misclassified points, thereby limiting the improvement in performance.

4.5 The Ability to Alleviate Imbalance Problem

To evaluate the effectiveness of our PALA in alleviating data imbalance, we record and analyze the accuracy and cross-entropy loss for head and tail classes across different data pairs with $\rho = 20$. Specifically, we normalize the accuracy and cross-entropy loss for each method to provide a clearer visualization. The results are shown in Figure 5. For the head class, the accuracy and corresponding cross-entropy loss ratios are similar across all three methods, indicating that their performance is comparable when data is sufficient. However, for the tail class, the performance of the other two methods
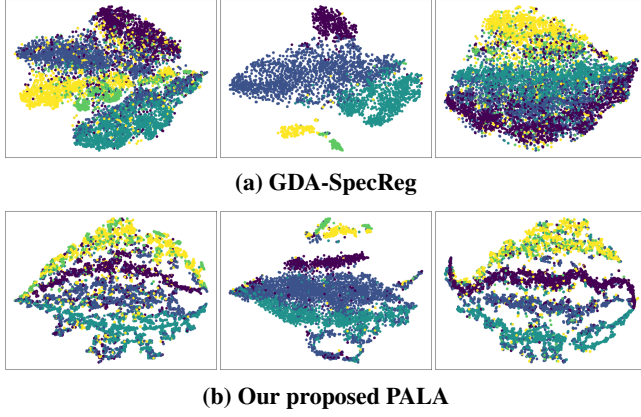
(a) GDA-SpecReg

(b) Our proposed PALA

Figure 6: Visualization of node embeddings learned by GDA-SpecReg and PALA via t-SNE. From left to right is balanced source graph, imbalanced source graph, and target graph, respectively.

significantly declines, highlighting their inadequate domain adaptation ability when handling data imbalance. Notably, GraphENS consistently outperforms GDA-SpecReg, likely due to its inherent capability to handle imbalanced data. In contrast, our method demonstrates stable performance for both head and tail classes, indicating its effectiveness in mitigating data imbalance and its strong generalization capability.

4.6 Visualization

To evaluate the domain adaptation ability of our PALA, we compare it with the competitive baseline GDA-SpecReg via visualization. With $\rho = 20$, we reduce node features encoded by the encoder to 2D using t-SNE, as shown in Figure 6. For each method, we present results for three scenarios: balanced source graph, imbalanced source graph, and target graph. In GDA-SpecReg, class boundaries in both source domains are unclear, and the target visualization differs significantly, suggesting poor transfer of domain-invariant features. In contrast, our PALA shows clear class boundaries, even though the imbalanced source differs slightly from the balanced one. The visualization of the target domain is similar to that of the balanced source domain, with clearer class boundaries, demonstrating that our method, by learning from the imbalanced source domain data, captures domain-invariant inter-class features and performs well in inter-class classification.

5 Related Work

5.1 Unsupervised Domain Adaption

Unsupervised Domain Adaption (UDA) aims at transferring knowledge from a labeled source domain to an unlabeled target domain, requiring domain adaptation methods to learn domain-invariant representations [Wang and Deng, 2018]. While successful in computer vision [Tzeng *et al.*, 2017], existing methods for i.i.d. data (e.g., image) are unsuitable for non-i.i.d. graph data. Recently, approaches for graph domain adaptation include adversarial learning and methods that minimize domain discrepancy based on a certain metric (e.g., MMD [Gretton *et al.*, 2012], subtree discrepancy [Wu *et al.*, 2023]. For the first group, ImGAGN [Qu *et*

al., 2021] balances class distribution by generating synthetic minority nodes and leveraging a GCN discriminator to address imbalanced graph classification. For the second group, GDASpecReg [You *et al.*, 2023] facilitates GNNs in capturing more cross-domain representations via spectral regularization. Similarly, A2GNN [Liu *et al.*, 2024a] adjusts propagation layers based on GNN Lipschitz bounds to enhance the ability of GNNs for cross-domain transfer. However, our proposed PALA further considers a more practical scenario—when the source graph is imbalanced—and proposes a novel prototype-anchored learning to eliminate prototype bias in the source graph, utilizing the learned prototypes for class-imbalanced graph domain adaptation.

5.2 Class-Imbalanced Learning on Graphs.

This line of work aims to address performance degradation caused by imbalanced data, primarily through three approaches: (i) Modifying the loss function to focus on underrepresented class [Cui *et al.*, 2019; Tan *et al.*, 2020; Menon *et al.*, 2021; Mao *et al.*, 2025], recent method like AutoLINC [Guo *et al.*, 2024] introduces an automated framework for optimizing loss functions to address class imbalance based on node distributions and graph topology. (ii) Post-hoc correction to adjust logits for tail class [Kang *et al.*, 2020; Hong *et al.*, 2021; Yi *et al.*, 2023]. (iii) Re-sampling to augment or generate tail class data [Wang *et al.*, 2021; Park *et al.*, 2021; Zhao *et al.*, 2021; Mao *et al.*, 2023; Huang *et al.*, 2023], among them, GraphSHA [Li *et al.*, 2023] enlarges decision boundaries for tail class by synthesizing harder samples. Additionally, BAT [Liu *et al.*, 2024c] mitigates bias in class-imbalanced node classification using a topological augmentation without rebalancing classes, while ImGCL [Zeng *et al.*, 2023] balances representations in graph contrastive learning with progressively balanced sampling and node centrality methods. However, distribution shifts in graphs are common in real-world scenarios, and addressing the class-imbalanced graph domain adaptation problem is a more challenging task. To this end, our proposed PALA learns recalibrated prototypes for each class to alleviate class imbalance bias, using the learned prototypes to guide pseudo-label assignment to target graph nodes and perform domain adaptation through weighted contrastive alignment.

6 Conclusion

In this paper, we highlight the class-imbalanced nature of the graph and propose a prototype-anchored learning and alignment framework termed PALA for the class-imbalanced graph domain adaption. Specifically, based on the encoded high-order structure information, we generate a set of categorical prototypes with corresponding proto-instances for recalibration and learning under imbalanced class distribution. Then, during the adaption phase, we introduce a robust contrastive prototype adaption strategy based on the generated pseudo label in the target graph domain. Experiments verify the effectiveness and superiority of our proposed PALA. In the future, we aim to integrate large language models into our prototype-anchored framework to further improve its adaptability and scalability, especially in handling more diverse and complex class-imbalanced graph data.

Contribution Statement

Acknowledgments

References

[Arpit *et al.*, 2017] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *Proceedings of the International Conference on Machine Learning*, pages 233–242, 2017.

[Cai *et al.*, 2024] Ruichu Cai, Fengzhu Wu, Zijian Li, Pengfei Wei, Lingling Yi, and Kun Zhang. Graph domain adaptation: A generative view. *ACM Transactions on Knowledge Discovery from Data*, 18(3):1–24, 2024.

[Chen *et al.*, 2019] Chaoqi Chen, Weiping Xie, Wenbing Huang, Yu Rong, Xinghao Ding, Yue Huang, Tingyang Xu, and Junzhou Huang. Progressive feature alignment for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 627–636, 2019.

[Cui *et al.*, 2019] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019.

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

[Graf *et al.*, 2021] Florian Graf, Christoph Hofer, Marc Niethammer, and Roland Kwitt. Dissecting supervised contrastive learning. In *Proceedings of the International Conference on Machine Learning*, pages 3821–3830, 2021.

[Gretton *et al.*, 2012] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[Guo *et al.*, 2024] Xinyu Guo, Kai Wu, Xiaoyu Zhang, and Jing Liu. Automated loss function search for class-imbalanced node classification. In *Forty-first International Conference on Machine Learning*, 2024.

[Hong *et al.*, 2021] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6626–6636, 2021.

[Huang *et al.*, 2016] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5375–5384, 2016.

[Huang *et al.*, 2023] Zhan Ao Huang, Yongsheng Sang, Yanan Sun, and Jiancheng Lv. Neural network with absent minority class samples and boundary shifting for imbalanced data classification. *Neural Computing and Applications*, 35(12):8937–8953, 2023.

[Ju *et al.*, 2024a] Wei Ju, Zhengyang Mao, Siyu Yi, Yifang Qin, Yiyang Gu, Zhiping Xiao, Yifan Wang, Xiao Luo, and Ming Zhang. Hypergraph-enhanced dual semi-supervised graph classification. In *Proceedings of the International Conference on Machine Learning*, 2024.

[Ju *et al.*, 2024b] Wei Ju, Siyu Yi, Yifan Wang, Qingqing Long, Junyu Luo, Zhiping Xiao, and Ming Zhang. A survey of data-efficient graph learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024.

[Ju *et al.*, 2024c] Wei Ju, Siyu Yi, Yifan Wang, Zhiping Xiao, Zhengyang Mao, Hourun Li, Yiyang Gu, Yifang Qin, Nan Yin, Senzhang Wang, et al. A survey of graph neural networks in real world: Imbalance, noise, privacy and ood challenges. *arXiv preprint arXiv:2403.04468*, 2024.

[Ju *et al.*, 2025] Wei Ju, Zhengyang Mao, Siyu Yi, Yifang Qin, Yiyang Gu, Zhiping Xiao, Jianhao Shen, Ziyue Qiao, and Ming Zhang. Cluster-guided contrastive class-imbalanced graph classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

[Kang *et al.*, 2020] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. In *International Conference on Learning Representations*, 2020.

[Kipf and Welling, 2017] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

[Li *et al.*, 2023] Wen-Zhi Li, Chang-Dong Wang, Hui Xiong, and Jian-Huang Lai. Graphsha: Synthesizing harder samples for class-imbalanced node classification. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1328–1340, 2023.

[Li *et al.*, 2025] Hourun Li, Yifan Wang, Zhiping Xiao, Jia Yang, Changling Zhou, Ming Zhang, and Wei Ju. Disco: graph-based disentangled contrastive learning for cold-start cross-domain recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 12049–12057, 2025.

[Liang *et al.*, 2020] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *Proceedings of the International Conference on Machine Learning*, pages 6028–6039, 2020.

[Liu *et al.*, 2020] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Proceedings of the Conference on Neural Information Processing Systems*, pages 20331–20342, 2020.

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

[Liu *et al.*, 2024b] Shikun Liu, Deyu Zou, Han Zhao, and Pan Li. Pairwise alignment improves graph domain adaptation. In *Forty-first International Conference on Machine Learning*, 2024.

[Liu *et al.*, 2024c] Zhining Liu, Ruizhong Qiu, Zhichen Zeng, Hyunsik Yoo, David Zhou, Zhe Xu, Yada Zhu, Kommy Weldemariam, Jingrui He, and Hanghang Tong. Class-imbalanced graph learning without class rebalancing. In *Forty-first International Conference on Machine Learning*, 2024.

[Luo *et al.*, 2023] Xiao Luo, Yusheng Zhao, Zhengyang Mao, Yifang Qin, Wei Ju, Ming Zhang, and Yizhou Sun. Rignn: A rationale perspective for semi-supervised open-world graph classification. *Transactions on Machine Learning Research*, 2023.

[Luo *et al.*, 2024a] Junyu Luo, Yiyang Gu, Xiao Luo, Wei Ju, Zhiping Xiao, Yusheng Zhao, Jingyang Yuan, and Ming Zhang. Gala: Graph diffusion-based alignment with jigsaw for source-free domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[Luo *et al.*, 2024b] Junyu Luo, Zhiping Xiao, Yifan Wang, Xiao Luo, Jingyang Yuan, Wei Ju, Langechuan Liu, and Ming Zhang. Rank and align: Towards effective source-free graph domain adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2024.

[Mao *et al.*, 2023] Zhengyang Mao, Wei Ju, Yifang Qin, Xiao Luo, and Ming Zhang. Rahnet: Retrieval augmented hybrid network for long-tailed graph classification. In *Proceedings of the 31st ACM international conference on multimedia*, pages 3817–3826, 2023.

[Mao *et al.*, 2025] Zhengyang Mao, Wei Ju, Siyu Yi, Yifan Wang, Zhiping Xiao, Qingqing Long, Nan Yin, Xinwang Liu, and Ming Zhang. Learning knowledge-diverse experts for long-tailed graph classification. *ACM Transactions on Knowledge Discovery from Data*, 19(2):1–24, 2025.

[Menon *et al.*, 2021] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. In *Proceedings of the International Conference on Learning Representations*, 2021.

[Park *et al.*, 2021] Joonhyung Park, Jaeyun Song, and Eunho Yang. Graphens: Neighbor-aware ego network synthesis for class-imbalanced node classification. In *International Conference on Learning Representations*, 2021.

[Qiao *et al.*, 2023] Ziyue Qiao, Xiao Luo, Meng Xiao, Hao Dong, Yuanchun Zhou, and Hui Xiong. Semi-supervised domain adaptation in graph transfer learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 2279–2287, 2023.

[Qiu *et al.*, 2021] Zhen Qiu, Yifan Zhang, Hongbin Lin, Shuaicheng Niu, Yanxia Liu, Qing Du, and Mingkui Tan. Source-free domain adaptation via avatar prototype generation and adaptation. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2021.

[Qu *et al.*, 2021] Liang Qu, Huaisheng Zhu, Ruiqi Zheng, Yuhui Shi, and Hongzhi Yin. Imgagn: Imbalanced network embedding via generative adversarial graph networks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data mining*, pages 1390–1398, 2021.

[Song *et al.*, 2022] Jaeyun Song, Joonhyung Park, and Eunho Yang. Tam: topology-aware margin loss for class-imbalanced node classification. In *International Conference on Machine Learning*, pages 20369–20383. PMLR, 2022.

[Tan *et al.*, 2020] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11662–11671, 2020.

[Tang *et al.*, 2008] 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*, pages 990–998, 2008.

[Tzeng *et al.*, 2017] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017.

[Veličković *et al.*, 2018] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[Wang and Deng, 2018] Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

[Wang *et al.*, 2021] Jianfeng Wang, Thomas Lukasiewicz, Xiaolin Hu, Jianfei Cai, and Zhenghua Xu. Rsg: A simple but effective module for learning imbalanced datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3784–3793, 2021.

[Wang *et al.*, 2024] Yifan Wang, Xiao Luo, Chong Chen, Xian-Sheng Hua, Ming Zhang, and Wei Ju. Disensemi: Semi-supervised graph classification via disentangled representation learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2024.

[Wu *et al.*, 2023] Jun Wu, Jingrui He, and Elizabeth Ainsworth. Non-iid transfer learning on graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10342–10350, 2023.

[Xu *et al.*, 2018] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2018.

[Yi *et al.*, 2023] Si-Yu Yi, Zhengyang Mao, Wei Ju, Yong-Dao Zhou, Luchen Liu, Xiao Luo, and Ming Zhang. Towards long-tailed recognition for graph classification via collaborative experts. *IEEE Transactions on Big Data*, 9(6):1683–1696, 2023.

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

[Yuan *et al.*, 2023] Jingyang Yuan, Xiao Luo, Yifang Qin, Zhengyang Mao, Wei Ju, and Ming Zhang. Alex: Towards effective graph transfer learning with noisy labels. In *Proceedings of the ACM International Conference on Multimedia*, pages 3647–3656, 2023.

[Zeng *et al.*, 2023] Liang Zeng, Lanqing Li, Ziqi Gao, Peilin Zhao, and Jian Li. Imgcl: Revisiting graph contrastive learning on imbalanced node classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11138–11146, 2023.

[Zhao *et al.*, 2021] Tianxiang Zhao, Xiang Zhang, and Suhang Wang. Graphsmote: Imbalanced node classification on graphs with graph neural networks. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 833–841, 2021.

[Zhuang and Ma, 2018] Chenyi Zhuang and Qiang Ma. Dual graph convolutional networks for graph-based semi-supervised classification. In *Proceedings of the Web Conference*, pages 499–508, 2018.