DANCE: Dual Unbiased Expansion with Group-acquired Alignment for Out-of-distribution Graph Fairness Learning

Yifan Wang [1 *] **Hourun Li** [2 *] **Ling Yue** [3] **Zhiping Xiao** [4 #] **Jia Yang** [3] **Changling Zhou** [3] **Wei Ju** [3 #]
Ming Zhang [2 #] **Xiao Luo** [5]

Abstract

Graph neural networks (GNNs) have shown strong performance in graph fairness learning, which aims to ensure that predictions are unbiased with respect to sensitive attributes. However, existing approaches usually assume that training and test data share the same distribution, which rarely holds in the real world. To tackle this challenge, we propose a novel approach named Dual Unbiased Expansion with Group-acquired Alignment (DANCE) for graph fairness learning under distribution shifts. The core idea of our DANCE is to synthesize challenging yet unbiased virtual graph data in both graph and hidden spaces, simulating distribution shifts from a data-centric view. Specifically, we introduce the unbiased Mixup in the hidden space, prioritizing minor groups to address the potential imbalance of sensitive attributes. Simultaneously, we conduct fairness-aware adversarial learning in the graph space to focus on challenging samples and improve model robustness. To further bridge the domain gap, we propose a group-acquired alignment objective that prioritizes negative pair groups with identical sensitive labels. Additionally, a representation disentanglement objective is adopted to decorrelate sensitive attributes and target representations for enhanced fairness. Extensive experiments demonstrate the superior effectiveness of the proposed DANCE.

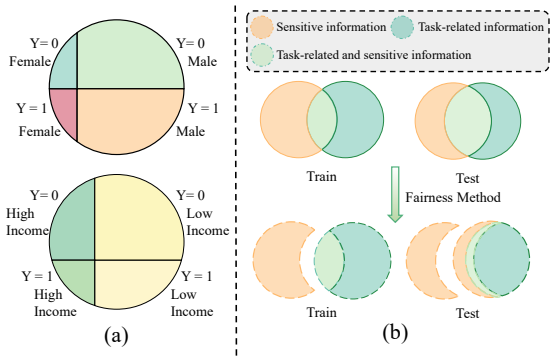


Figure 1. An illustration of sensitive group imbalance (Gender and Income as sensitive attributes) and inevitably removing task-related information challenges under distribution shifts.

1. Introduction

Graph-based machine learning methods, especially Graph Neural Networks (GNNs), have been adopted as the *de facto* approach for prediction tasks on graph-structured data (Wu et al., 2020; Ju et al., 2024). Among these tasks, node classification (Kipf & Welling, 2017) is fundamental, which endeavors to predict the category of each node within a graph and has been widely applied in various applications, i.e., community detection (Ren et al., 2025), molecular property prediction (Zhuang et al., 2023) and cross-modal retrieval (Li et al., 2024a). However, graph learning often suffers from fairness issues due to inherent biases in the training data, with the message-passing mechanism in GNNs further amplifying these biases (Dong et al., 2023).

To mitigate this issue, significant efforts have been devoted in recent years to achieving fair graph representation learning. Depending on the stage at which fairness interventions are applied, these efforts are typically grouped into pre-, in- and post-processing strategies. Pre-processing strategies address biases prior to the model training stage by modifying the input graph data, such as by applying node feature masking (Köse & Shen, 2021; Wang et al., 2024), edge perturbation (Li et al., 2021; Spinelli et al., 2021), mixup (Li et al., 2024c) or distribution alignment (Dong et al., 2022). In contrast, in-processing strategies incorporate fairness constraints during the training stage by modifying learning ob-

[*]Equal contribution [1]School of Information Technology & Management, University of International Business and Economics, Beijing, China [2]State Key Laboratory for Multimedia Information Processing, School of Computer Science, PKU-Anker LLM Lab, Peking University, Beijing, China [3]Peking University, Beijing, China [4]Paul G. Allen School of Computer Science and Engineering, University of Washington, Seattle, WA, USA [5]Department of Computer Science, University of California, Los Angeles, USA. Correspondence to: Zhiping Xiao <patxiao@uw.edu>, Wei Ju <juwei@pku.edu.cn>, Ming Zhang <mzhang_cs@pku.edu.cn>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

jectives, such as through regularization (Bose & Hamilton, 2019), contrastive learning (Zhang et al., 2025) and adversarial training (Ling et al., 2023). Finally, post-processing strategies modify the model's output after training to mitigate bias and enhance fairness (Dai & Wang, 2021).

However, most existing fairness-aware learning works are built on the assumption that training and test nodes in the graph are independently and identically distributed (i.i.d.), which rarely holds in real-world scenarios. In practice, nodes often belong to multiple environments, leading to distribution shifts caused by complex data generation processes (Bengio et al., 2019; Li et al., 2023a). Therefore, these approaches often suffer from poor performance and unstable predictions (Li et al., 2022a; Zhang et al., 2024b) when adapting to new environments (groups) (Pham et al., 2023; Li et al., 2024b). To address these challenges, significant efforts have been devoted to graph out-of-distribution (OOD) generalization under distribution shifts. Recent advancements include techniques such as data augmentation (Han et al., 2022; Sui et al., 2023), (causal) invariant learning (Li et al., 2022b; Yu et al., 2023) and model architecture designs (Yang et al., 2023).

Despite significant progress, we argue that achieving fair graph learning under the distribution shift remains challenging due to the following two key issues: ❶ *Sensitive group imbalance arising from diverse graph distributions.* Sensitive groups are often imbalanced in the graph, with the generalization process under the distribution shift in the graph being highly skewed. This imbalance forces the model to focus primarily on the majority group, which results in insufficient learning of the minority group to extract fair representation. ❷ *Inevitably removing task-related information across environments.* Graph distribution shift often introduces unintended correlation between the target and sensitive information in practice. For example, prior studies have observed that a fair income prediction model trained in one state may lose its fairness when applied to another state (Ding et al., 2021). When this correlation is entangled, it will lead to a conflict between the fairness and performance objectives of the task.

To this end, we propose a novel approach in this paper, named Dual Unbiased Expansion with Group-acquired Alignment (DANCE), which aims to address the challenges of sensitive group imbalance and the fairness-performance conflict under graph distribution shifts. Specifically, since the test graph distribution is unknown, we investigate graph data expansion in both graph structural and feature space, generating OOD graph data to expand the training distribution and enhance generalization. We first synthesize harder minor samples from the structural aspect to enlarge the squeezed sensitive group. Then, we focus on the feature aspect and employ an adversarial module to generate the graph with significant bias. Based on the expanded graph, we introduce group-acquired alignment to ensure fair node representations, which prioritizes sample pairs with identical sensitive labels as negatives. In addition, we explicitly disentangle sensitive and task-related information to ensure that task-related information remains invariant across distribution shifts. Experimental results on multiple real-world datasets show that our DANCE achieves superior fairness and performance compared to state-of-the-art methods.

In summary, the primary contributions of the paper are as follows: ❶ *New Perspective.* We study an underexplored yet practical problem of out-of-distribution graph fairness learning and propose a data-centric view to solve the problem. ❷ *Novel Methodology.* Our DANCE not only generates challenging but unbiased virtual graph data to simulate the distribution shift, but also introduces a group-acquired alignment to minimize the domain gap with fairness considered. ❸ *Extensive Experiments.* We conduct comprehensive experiments on multiple real-world datasets to evaluate the framework. The results demonstrate that the proposed DANCE achieves superior performance and fairness under distribution shifts. The code is available at `https://github.com/HourunLi/DANCE_ICML_2025`.

2. Preliminaries & Problem Definition

Notations. Let a graph be represented as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \boldsymbol{X})$, in which $\mathcal{V} = \{v_1, \ldots, v_N\}$ denotes the node set containing N nodes, and $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$ is the edge set. We use the adjacency matrix $A \in \{0,1\}^{N \times N}$ to describe the connectivity and structural dependencies of nodes in the graph, where $\boldsymbol{A}_{uv} = 1$ if $(u, v) \in \mathcal{E}$, otherwise $\boldsymbol{A}_{uv} = 0$. The node feature matrix can be represented as $\boldsymbol{X} \in \mathbb{R}^{N \times d}$, where row $\boldsymbol{X}_v \in \mathbb{R}^d$ denote the feature vector of the node v with dimension d. The sensitive attributes of the node set are specified by the t-th dimension of $\boldsymbol{X}$, i.e., $\boldsymbol{S} = \boldsymbol{X}_{:,t} = \{s_1, \ldots, s_N\} \in \{0,1\}^N$, where s_v is the sensitive attribute associated with node v. We consider the binary node classification task, with the node label matrix denoted as $\boldsymbol{Y} \in \{0,1\}^N$.

Definition 2.1 (**Sensitive Group**)**.** The sensitive group denotes the node set divided by the sensitive attribute:

$$\mathcal{V}_s = \{v \in \mathcal{V} | s_v = s\}. \tag{1}$$

Definition 2.2 (**EO Group**)**.** The Equality Odds (EO) group of the graph is divided by sensitive attribute s and label y:

$$\mathcal{V}_s^y = \{v \in \mathcal{V} | (s_v = s) \cap (y_v = y)\}. \tag{2}$$

Definition 2.3 (**Demographic Parity**)**.** Demographic parity (Calders et al., 2009) stipulates that different demographic groups should have equal positive prediction probabilities. Accordingly, Δ_{DP} can be defined as:

$$\Delta_{DP} = |\mathbb{E}_{u \in \mathcal{V}}(\hat{y}_u = 1 | s_u = 1) - \mathbb{E}_{v \in \mathcal{V}}(\hat{y}_v = 1 | s_v = 0)|, \tag{3}$$

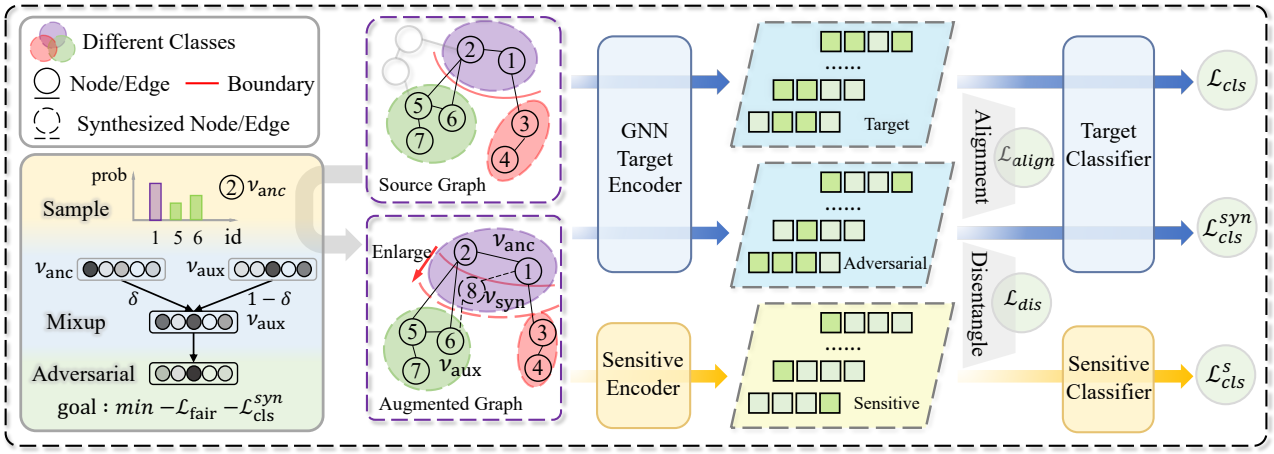


Figure 2. Illustration of DANCE, which contains three modules: (1) **Graph Expansion for Distribution Shifts**: Generate unbiased virtual graph in both graph and hidden space. (2) **Group-acquired Alignment**: We introduce group alignment to prioritize negative pair groups with identical sensitive labels. (3) **Representation Disentanglement**: We decorrelate sensitive attributes for enhanced fairness.

where the predicted and ground-truth labels for the node v are $\hat{y}_v$ and y_v. The independence between predictions $\hat{y}$ and sensitive qualities s, namely $\hat{y} \perp s$, is quantified by Δ_{DP}.

Definition 2.4 (Equalized Odds). Equal odds (Hardt et al., 2016) seeks to enforce that both the True Positive Rate (TPR) and False Positive Rate (FPR) should be equal in various demographic groups. And Δ_{EO} can be written as:

$$\Delta_{EO} = \frac{1}{2}\sum_{y=0}^{1} |\mathbb{E}_{u\in\mathcal{V}}(\hat{y}_u = y | y_u = y, s_u = 1) - \mathbb{E}_{v\in\mathcal{V}}(\hat{y}_v = y | y_v = y, s_v = 0)|, \tag{4}$$

where the independence between the $\hat{y}$ and s conditions on the ground-truth y, namely $\hat{y} \perp s | y$, is quantified by Δ_{EO}.

Problem Definition. We denote the generation process of a graph as $P(\boldsymbol{A}, \boldsymbol{X}, \boldsymbol{Y}|e) = P(\boldsymbol{A}, \boldsymbol{X}|e)P(\boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{X}, e)$, where $e \in \mathcal{E}$ represents the environment in the whole environment set $\mathcal{E}$. The structure and feature of graph is generated by $P(\boldsymbol{A}, \boldsymbol{X}|e)$ and node labels are determined by $P(\boldsymbol{Y}|\boldsymbol{A}, \boldsymbol{X}, e)$. The distribution shifts of the graph induce that $P_{tr}(\boldsymbol{A}_v, \boldsymbol{X}_v) \neq P_{te}(\boldsymbol{A}_v, \boldsymbol{X}_v)$, i.e., the graph distributions generating the ego-graphs $\mathcal{G}_v = (\boldsymbol{A}_v, \boldsymbol{X}_v)$ and labels $\boldsymbol{Y}_v$ for the training and test nodes are different. For example, the diverse communities that are partitioned within the graph. The objective is to learn a powerful GNN $f_\theta^*(\cdot) = h^*(g^*(\cdot))$ that achieves both good performance and fairness over all environments for node classification task:

$$f^* = \arg\min_{f_\theta} \sup_{e\in\mathcal{E}} \mathcal{R}(f|e), \tag{5}$$

where $f^*(\cdot)$ is decomposed into two parts, with graph encoder $g(\cdot)$ and classifier $h(\cdot)$. And $\mathcal{R}(f|e)$ represents the empirical risk that ensures both good performance and fairness for the nodes within environment e.

3. Methodology

This paper studies the problem of out-of-distribution graph fairness learning and a novel framework DANCE is proposed to solve this problem. The high-level idea of our DANCE is to expand the graph data in both graph and feature space, which can simulate the distribution shifts in a data-centric view. On the one hand, we utilize unbiased Mixup to enhance the attribute balance in the expanded dataset. On the other hand, we adopt fairness-aware adversarial learning to generate challenging samples for robustness. These generated virtual data are incorporated into a group-acquired alignment framework to emphasize negative pairs with identical sensitive labels for enhanced fairness. We also leverage representation disentanglement to facilitate the decorrelation of sensitive and target information. An overview of our proposed DANCE is illustrated in Figure 2. More details of the framework are introduced as below.

3.1. Dual Graph Expansion for Simulating Distribution Shifts

Unbiased Mixup for Balanced Sensitive Attributes. Since the sensitive groups of nodes in the graph are often imbalanced, we are motivated to enlarge the minor group boundary. Specifically, we employ the confidence (Wang et al., 2021) to calculate the node hardness in the minor sensitive group, which can be defined as:

$$\boldsymbol{c}_v = 1 - \sigma(y_v, \hat{y}_v),\ \hat{y}_v = f(\boldsymbol{A}_v, \boldsymbol{X}_v), \tag{6}$$

where $\sigma(y_v, \hat{y}_v) = y_v\hat{y}_v + (1 - y_v)(1 - \hat{y}_v)$, y_v and $\hat{y}_v$ denote the ground-truth and predicted label of node v respectively, and we calculate $\hat{y}_v$ from the previous epoch in practice. Then, we can identify the minor anchor node v_{anc} from a multinomial distribution with the node hard-

ness $\boldsymbol{c} \in \mathbb{R}^N$ as the input to calculate probability. Then, we sample from another multinomial distribution to identify auxiliary node v_{aux} based on the confidence in class $y_{v_{anc}}$, i.e., $\boldsymbol{c}_v = 1 - \sigma(y_{v_{anc}}, \hat{y}_v)$. We aim to enlarge the boundary of the minor sensitive group while avoiding degenerating the other groups in the graph. Thus, we employ a simple Mixup (Zhang et al., 2018) strategy to generate the synthesized node feature, which can be defined as:

$$\boldsymbol{X}_{syn} = \delta \boldsymbol{X}_{anc} + (1-\delta)\boldsymbol{X}_{aux}, \tag{7}$$

where δ is the trade-off parameter, and as δ decreases, the generated synthesized node features become more similar to the auxiliary node, thereby expanding the boundary of the minor sensitive group. Note that edges dominate the message-passing procedures during GNN propagation. Here, we aim for additional synthesized edges for message passing beyond the minority group boundary while blocking propagation to the other groups, preserving the degeneration of the sensitive groups. Instead of connecting v_{syn} to both $\mathcal{G}_{anc}$ and $\mathcal{G}_{aux}$, we only connect with meaningful neighborhoods in $\mathcal{G}_{anc}$. Specifically, we define the graph diffusion (Gasteiger et al., 2019) process on the graph:

$$\boldsymbol{S}_{anc} = \sum_{r=0}^{\infty} \theta_r \boldsymbol{T}_{anc}^r, \ \boldsymbol{T}_{anc} = \boldsymbol{A}_{anc}\boldsymbol{D}_{anc}^{-1}, \tag{8}$$

where $\theta_r = \alpha(1-\alpha)^r$ for Personalized PageRank (PPR) and $\theta_r = e^{-t}\frac{t^r}{r!}$ for Heat Kernel (HK) examples, D_{anc} denote the degree matrix of ego-graph. We can utilize the sparse output of $\boldsymbol{S}_{anc}$ to form a multinomial distribution for sampling neighbors of v_{syn}, with the number of neighbors determined based on the entire graph to preserve the degree statistics (Li et al., 2023b). The synthesized edges and labels of the synthesized node can be defined as:

$$\mathcal{N}_{syn} \sim P_{1hop}^{diff}(v_{anc}), \ y_{syn} = y_{anc}, \tag{9}$$

where $P_{1hop}^{diff}(v_{anc})$ is the 1-hop neighbors distribution. Note that we generate the synthesized node in the graph w.r.t. ratio r, namely, we add the synthesized minor nodes from N_{min} to $(1-\gamma) * N_{min} + \gamma * N_{max}$, where N_{min} and N_{max} denote the number of nodes in two sensitive groups.

Adversarial Learning for Challenging Graph Data. To further expand the graph in the feature space, we consider the worst-case shift near the training graph data. Given the synthesized graph $\mathcal{G}' = (\boldsymbol{A}', \boldsymbol{X}')$, we feed it into $f(\cdot)$ to get the predicted label $\hat{y}'_v$. The classification loss can be:

$$\mathcal{L}_{cls}^{syn} = -\sum_{v \in \mathcal{V}_{tr}} \mathrm{BCE}(y_v, \hat{y}'_v)), \tag{10}$$

where $\mathrm{BCE}(\cdot)$ denotes the binary cross-entropy loss. At regular intervals during training, we utilize an MLP-based generator $\rho_\psi(\cdot) : \boldsymbol{X}' \to \boldsymbol{X}_\rho$ to identify node features that contribute to poor fairness performance. By leveraging adversarial training (Volpi et al., 2018; Liao et al., 2021), the feature generator is optimized to simultaneously maximize both the fairness loss and the classification loss:

$$\begin{aligned} &\max_{\psi}(\mathcal{L}_{cls}^{syn} + \mathcal{L}_{fair} - \lambda\|\boldsymbol{X}' - \boldsymbol{X}_\rho\|_F^2), \\ &\mathcal{L}_{fair} = \frac{1}{2}\sum_{y=0}^{1} |\mathbb{E}_{u \in \mathcal{V}'^y_1}(\hat{y}_u = y|\rho(x_u), \boldsymbol{A}') - \\ &\qquad \mathbb{E}_{u \in \mathcal{V}'^y_0}(\hat{y}_u = y|\rho(x_u), \boldsymbol{A}')|, \end{aligned} \tag{11}$$

where $\mathcal{V}'^y_s$ denote the EO groups in graph $\mathcal{G}'$. Note that we additionally incorporate a regularization term to guarantee that the generated features do not deviate significantly from those of the training graph.

3.2. Group-acquired Alignment for Graph Heterogeneity Reduction

To ensure the model generates a consistent and fair representation under graphs with significant biases, we disentangle the node similarities into fine-grained types for demographic group alignment, explicitly penalizing the model that learns unwanted information. Specifically, we treat the input and generated graphs as two views and encode them to obtain the node representations, i.e., $\boldsymbol{Z}^* \in \mathbb{R}^{N \times d'}$ ($* \in \{1, 2\}$) with dimension d'. We adopt different types of similarities between the anchor node v and other nodes in the graph to construct the sample set, which can help us distinguish positive (negative) pairs with different relationships of sensitive labels (Park et al., 2022; Zhang et al., 2024a). There are four groups with the following definition. ❶ *Intra-Group* (***IG***): The similarity is defined within the EO group, and the node set can be defined as $\boldsymbol{Z}_{ig}^*(v) = \{\boldsymbol{z}_{ig} \in \boldsymbol{Z}^* | s_{ig} = s_v \cap y_{ig} = y_v\}$. ❷ *Sensitive Inter-Group* (***SG***): The similarity is defined between an anchor and nodes from the same target class but with different sensitive attributes, and the node set can be written as $\boldsymbol{Z}_{sg}^*(v) = \{\boldsymbol{z}_{sg} \in \boldsymbol{Z}^* | s_{sg} \neq s_v \cap y_{sg} = y_v\}$. ❸ *Target Inter-Group* (***TG***): We focus on the similarity between an anchor and nodes from the same sensitive attribute, which are labeled with different target classes, and the node set can be defined as $\boldsymbol{Z}_{tg}^*(v) = \{\boldsymbol{z}_{tg} \in \boldsymbol{Z}^* | s_{tg} = s_v \cap y_{tg} \neq y_v\}$. ❹ *Target & Sensitive Inter-Group* (***TSG***): The similarity is defined between an anchor and nodes that differ in both sensitive attributes and target classes. The node set is defined as $\boldsymbol{Z}_{tsg}^*(v) = \{\boldsymbol{z}_{tsg} \in \boldsymbol{Z}^* | s_{tsg} \neq s_v \cap y_{tsg} \neq y_v\}$.

Based on this, we encourage the similarity in *IG* and *SG* are larger than *TG* and *TSG* between two views for alignment:

$$\begin{aligned} \mathcal{L}_{align} = -\sum_{\forall y,s} \frac{1}{|\boldsymbol{Z}_s^{*,y}|} \sum_{\boldsymbol{z}_v \in \boldsymbol{Z}_s^{*,y}} \sum_{\forall s} \frac{1}{|\boldsymbol{Z}_{p,s}^*(v)|} \\ \sum_{\boldsymbol{z}_p \in \boldsymbol{Z}_{p,s}^*(v)} \log \frac{\phi_p}{\sum_{\boldsymbol{z}_{tg} \in \boldsymbol{Z}_{tg}^*(v)} \phi_{tg}}, \end{aligned} \tag{12}$$

where $\boldsymbol{Z}_s^{*,y} = \{\boldsymbol{z}_v \in \boldsymbol{Z}^* | y_v = y, s_v = s\}$, $\boldsymbol{Z}_{p,s}^*(v) = \{\boldsymbol{z}_p \in \boldsymbol{Z}_{ig}^*(v) \cup \boldsymbol{Z}_{sg}^*(v) | s_p = s\}$, and $\phi_{tg} = \exp(\boldsymbol{z}_v \cdot \boldsymbol{z}_{tg}/\tau)$. Note that since the number of anchors and positive samples is imbalanced across sensitive groups, we follow prior work (Park et al., 2022) and apply group-wise normalization to mitigate the unfairness caused by these disparities.

3.3. Representation Disentanglement for Enhanced Fairness

To further prevent the conflict between fairness and performance during graph distribution shifts, we employ another GNN to extract node representation $\boldsymbol{Z}^-$ of the augmented graph and cast it as the additional sensitive-aware negative view. Toward this end, we use a sensitive discriminator $\xi(\cdot) : \boldsymbol{Z}^- \to \boldsymbol{S}$ and the classification loss can be:

$$\mathcal{L}_{cls}^s = -\sum_{v \in \mathcal{V}_{tr}} \text{BCE}(s_v, \xi(\boldsymbol{z}_v^-)). \tag{13}$$

Based on the negative view, we utilize a node representation disentanglement loss to explicitly decorrelate sensitive information (Park & Byun, 2024), defined as follows:

$$\mathcal{L}_{dis} = -\sum_{\forall y,s} \frac{1}{|\boldsymbol{Z}_s^{-,y}|} \sum_{\boldsymbol{z}_v \in \boldsymbol{Z}_s^{-,y}} \sum_{\forall s} \frac{1}{|\boldsymbol{Z}_{ig,s}^-(v)|} \sum_{\boldsymbol{z}_p \in \boldsymbol{Z}_{ig,s}^-(v)} \log \frac{\phi_p}{\sum_{\boldsymbol{z}_{ig} \in \boldsymbol{Z}_{ig}^2(v)} \phi_{ig}}, \tag{14}$$

where $\boldsymbol{Z}_{ig,s}^-(v) = \{\boldsymbol{z}_p \in \boldsymbol{Z}_{ig}^-(v) | s_p = s\}$ and $\phi_{ig} = \exp(\boldsymbol{z}_v \cdot \boldsymbol{z}_{ig}/\tau)$. Here, we encourage the augmented graph and its negative view to contain distinct information by reducing the similarity within the *IG* group between $\boldsymbol{Z}^2$ and $\boldsymbol{Z}^-$.

3.4. Overall Optimization

We also train the GNN model $f(\cdot)$ on the input graph to ensure the node classification accuracy, defined as:

$$\mathcal{L}_{cls} = -\sum_{v \in \mathcal{V}_{tr}} \text{BCE}(y_v, \hat{y}_v)), \tag{15}$$

where $\hat{y}_v$ is the predicted label. The overall optimizing process of DANCE can be formulated as:

$$\begin{cases} \min_\theta \mathcal{L}_{cls} + \mathcal{L}_{cls}^s + \mathcal{L}_{cls}^{syn} + \beta(\mathcal{L}_{align} + \mathcal{L}_{dis}), \\ \min_\psi -\mathcal{L}_{cls}^{syn} - \mathcal{L}_{fair} + \lambda \|\boldsymbol{X}' - \boldsymbol{X}_\rho\|_F^2, \end{cases} \tag{16}$$

where λ denotes the weight of the regularization term.

3.5. Theoretical Analysis

Here, we prove that the graph diffusion process we put forward can precisely control the information propagation between different groups. Additionally, the convergence proof of the loss function is shown.

To begin, we introduce the following notations. Let $\mathcal{V}_s$ and $\mathcal{V}_{-s} = \mathcal{V} \setminus \mathcal{V}_s$ be the set of minority and non-minority group nodes, respectively. The adjacency matrix of the graph with synthetic edges is $\boldsymbol{A}_{anc}$, and the corresponding degree matrix is $\boldsymbol{D}_{anc}$, where $\boldsymbol{D}_{anc,ii} = \sum_{j=1}^{|\mathcal{V}|} \boldsymbol{A}_{anc,ij}$. Let $\boldsymbol{H}^{(l)} \in \mathbb{R}^{|\mathcal{V}| \times d}$ be the representation matrix of all nodes in the l-th layer, where $Z = \boldsymbol{H}^{(1)} \| \ldots \| \boldsymbol{H}^{(L)}$ denotes the concatenated representation output from all L layers. The information propagation rule is:

$$\boldsymbol{H}^{(l)} = \sigma\left(\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)} \boldsymbol{W}^{(l)}) + \boldsymbol{B}^{(l)}\right), \tag{17}$$

where σ denotes the activation function. $\boldsymbol{W}^{(l)}$ and $\boldsymbol{B}^{(l)}$ refers to the learnable weight and bias matrix for the l-th layer. Let ω be the model parameter, which can represent either the classifier parameter θ or the generator parameter ψ. Define the loss function such that when $\omega = \theta$,

$$\mathcal{F}(\omega) = \mathcal{L}_1(\theta) = \mathcal{L}_{cls} + \mathcal{L}_{cls}^s + \mathcal{L}_{cls}^{syn} + \beta(\mathcal{L}_{align} + \mathcal{L}_{dis}). \tag{18}$$

When $\omega = \psi$,

$$\mathcal{F}(\omega) = \mathcal{L}_2(\psi) = -\mathcal{L}_{cls}^{syn} - \mathcal{L}_{fair} + \lambda \|\boldsymbol{X}' - \boldsymbol{X}\|_F^2. \tag{19}$$

Theorem 3.1. *Given the above notations, for any $i \in \mathcal{V}_s$, there exists $j \in \mathcal{V}_{-s}$ such that*

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} \neq 0; \tag{20}$$

For any $i \in \mathcal{V}_{-s}$, for all $j \in \mathcal{V}_s$, we have

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} = 0. \tag{21}$$

Theorem 3.1 indicates that the graph diffusion method can precisely control the information propagation between different groups, which is crucial for improving the fairness and performance of the model when dealing with data from different groups.

Theorem 3.2. *Suppose that the loss function $\mathcal{F}(\omega)$ is differentiable with respect to the parameter ω, and there exists a constant $M > 0$ such that the gradient of the loss function is M-Lipschitz continuous. We adopt the SGD algorithm to update the parameter. When an appropriate learning rate α is chosen such that $0 < \alpha < \frac{2}{M}$, we have $\lim_{t\to\infty} \|\nabla \mathcal{F}(\omega_t)\| = 0$ and $\lim_{t\to\infty} \mathcal{F}(\omega_t)$ exists, which means the loss function $\mathcal{F}(\omega)$ converges.*

The Lipschitz continuity assumption limits the rate of change of the gradient of the loss function, which is a common condition for the convergence proof of existing optimization algorithms (Meng et al., 2024).

Table 1. Classification and fairness metrics (%±σ) on Credit-Cs. ↑ indicates that higher values are better, while ↓ indicating the opposite. We highlight the best results in **bold**.

Dataset	Metric	MLP	GCN	FairVGNN	NIFTY	EDITS	EERM	CAF	SAGM	RFR	FatraGNN	DANCE
C1	ACC↑	75.69±5.64	76.69±2.48	77.45±0.31	77.51±0.03	77.06±0.03	77.04±0.09	76.54±0.81	77.16±0.45	76.83±1.26	77.31±0.1	**78.58±0.41**
	ROC-AUC↑	64.38±0.47	65.54±1.43	67.07±0.4	68.43±0.25	65.25±1.29	66.54±1.37	67.58±1.36	65.35±1.03	66.08±0.81	65.41±1.28	**72.18±0.08**
	Δ_{DP} ↓	5.2±14.92	7.4±6.69	1±0.63	4.34±0.03	2.43±0.03	5.46±0.38	5.61±0.84	5.21±0.76	3.46±0.98	**0.50±0.21**	0.74±0.56
	Δ_{EO} ↓	7.92±15.86	6.31±6	0.85±0.2	2.73±0.04	3.24±0.03	6.47±0.26	5.03±1.49	4.58±0.68	3.19±0.73	0.71±0.03	**0.70±0.49**
C2	ACC↑	72.05±2.6	75.42±0.44	75.49±1.7	74.44±0.47	77.07±0.22	76.16±0.91	75.49±0.82	76.38±0.91	75.24±0.68	77.12±0.28	**78.4±0.62**
	ROC-AUC↑	62.36±6.29	63.76±3.07	63.8±0.25	60.63±10.06	62.5±3.27	65.49±2.39	62.36±1.18	62.65±0.89	61.35±0.92	64.16±0.69	**75.5±0.21**
	Δ_{DP} ↓	8.14±4.39	8.74±3.6	3.54±0.42	3.54±1.6	2.98±0.01	4.22±1.38	3.61±0.74	5.34±0.67	2.35±0.39	1.64±1.06	**1.62±1.53**
	Δ_{EO} ↓	6.7±4.3	7.35±2.64	2.63±0.61	2.34±0.63	3.65±0.16	5.71±1.1	3.57±0.89	4.37±0.98	2.96±0.81	0.95±0.7	**0.40±0.08**
C3	ACC↑	68.15±0.16	70.31±1.79	71.49±0.56	70.11±0.04	70.89±0.96	71.43±1.24	71.28±1.35	70.83±0.65	70.03±0.48	71.81±0.39	**72.77±0.73**
	ROC-AUC↑	64.64±4.49	65.90±1.72	65.96±0.19	64.75±0.14	63.18±2.53	65.36±1.03	64.37±1.06	64.52±0.91	63.59±0.63	65.7±0.91	**75.63±0.12**
	Δ_{DP} ↓	8.7±0.12	9.46±10.06	3.05±1.76	3.54±0.07	3.22±0.45	5.63±9.43	5.28±0.94	5.12±1.36	2.86±1.25	**0.25±0.2**	1.38±1.18
	Δ_{EO} ↓	9.47±0.03	9.71±8.29	3.35±2.46	2.23±0.08	1.87±0.36	5.34±4.62	4.82±1.33	5.57±0.79	3.41±0.89	0.81±0.56	**0.31±0.04**
C4	ACC↑	68.26±3.09	70.89±5.38	71.74±0.45	71.84±6.36	71.28±0.2	71.35±4.28	71.48±0.81	71.22±0.29	71.47±0.92	72.15±0.42	**72.19±0.71**
	ROC-AUC↑	65.32±4.42	64.28±3.45	66.45±1.61	66.98±2.3	63.45±0.6	64.04±0.67	66.59±1.24	63.45±0.88	65.27±0.68	67.66±0.87	**77.56±0.02**
	Δ_{DP} ↓	7.46±12.08	6.13±4.08	3.46±0.05	7.84±9.64	3.42±0.47	4.35±0.85	3.73±0.92	5.46±1.18	3.59±0.41	**0.61±0.08**	1.24±1.25
	Δ_{EO} ↓	6.61±11.22	8.16±2.38	2.82±0.19	2.18±9.91	3.22±0	5.07±0.74	4.18±0.85	5.34±0.22	2.46±0.87	1.16±0.13	**0.36±0.20**
	Rank	11	10	3	5	4	8	7	9	6	2	1

Table 2. Classification and fairness metrics (%±σ) on Pokecs. ↑ indicates that higher values are better, while ↓ indicating the opposite. We highlight the best results in **bold**.

Dataset	Metric	MLP	GCN	FairVGNN	NIFTY	CAF	SAGM	RFR	FatraGNN	DANCE
Pokec-n	ACC↑	52.74±3.67	54.83±2.34	60.8±0.54	58.68±5.54	59.37±1.45	58.78±2.33	57.42±3.68	62.00±0.24	**66.58±0.20**
	ROC-AUC↑	65.38±0.43	63.48±2.34	65.26±1.45	67.09±2.25	66.86±1.32	65.67±2.45	65.29±1.36	67.82±3.23	**72.88±0.10**
	Δ_{DP} ↓	4.86±1.23	7.38±0.28	5.88±2.34	4.21±3.43	5.49±2.65	5.67±3.22	4.56±2.85	1.34±0.27	**0.83±0.22**
	Δ_{EO} ↓	4.16±2.34	6.37±0.52	6.26±2.21	3.82±3.88	5.02±0.73	4.19±2.45	3.41±2.37	1.43±2.68	**0.29±0.12**
	Rank	8	9	7	3	4	6	5	2	1

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate the performance of DANCE under distribution shifts using three real-world datasets, described as follows: 1) *Credit-Cs* is derived from the Credit dataset (Yeh & Lien, 2009), where nodes represent credit card users. The task involves classifying users' credit risk as either high or low, with "age" designated as the sensitive attribute. To explore the data structure, we apply a modularity-based community detection method (Newman, 2006), partitioning Credit-Cs into five communities labeled C0 through C4. These partitions reveal different data distributions across the communities. Community C0 is utilized for training and validation, and the remaining communities (C1 to C4) serve as the test set. 2) *Pokecs* is constructed from the Slovakian social network (Dai & Wang, 2022), where users are grouped by province. The task is to predict users' working fields, with "region" serving as the sensitive attribute. Pokecs consists of two graphs, Pokec-z and Pokec-n, each containing users from two different regions. Pokec-z is used for training and validation, while Pokec-n serves as the test set. 3) *Bail-Bs* is derived from a fairness-oriented graph Bail, in which the nodes represent defendants released on bail (Jordan & Freiburger, 2015). The task aims to determine whether a defendant should be granted bail, with "race" as the sensitive attribute. Following the approach used for Credit-Cs, Bail-Bs is partitioned into five communities, labeled B0 through B4. B0 is used for training and validation, while B1–B4 serve as the test set.

Baselines. We compare DANCE with four baseline groups: (A) Traditional learning methods: Foundational representation learning approaches, including MLP (Pal & Mitra, 1992), GCN (Kipf & Welling, 2016). (B) Fair GNNs: GNNs designed to improve fairness, including FairVGNN (Wang et al., 2022), NIFTY (Agarwal et al., 2021), EDITS (Dong et al., 2022), CAF (Guo et al., 2023). (C) Out-of-distribution (OOD) methods: Approaches for learning robust representations that generalize under distribution shifts, including EERM (Wu et al., 2022), SAGM (Wang et al., 2023). (D) Fairness under distribution shifts methods: Methods addressing distribution shifts while preserving fairness across training and test distributions, such as RFR (Jiang et al., 2023), FatraGNN (Li et al., 2024b).

Performance Evaluation. We evaluate node classification performance using two key metrics: accuracy (ACC) and ROC-AUC. Fairness is measured with Δ_{DP} and Δ_{EO}, as detailed in Section 2, where lower values indicate better fairness. To comprehensively assess both classification and fairness, we adopt the combined metric proposed in FatraGNN (Li et al., 2024b), denoted as $c = \text{ACC} + \text{ROC\_AUC} - \Delta_{DP} - \Delta_{EO}$, where higher values indicate better overall performance. The final score of each method is calculated by summing its scores across all test graphs, based on which the overall rankings are provided for comparison.

Experimental Setting. In the experiments, we perform hyperparameter tuning via grid search across all dataset groups to ensure fair and comprehensive evaluation. For DANCE, the embedding dimension is set to 256. We explore the number of graph encoder layers in the range of $[1, 5]$, dropout rates between $[0, 0.5]$, and learning rates in $[0, 0.005]$. Additionally, the trade-off parameter δ in the Mixup strategy is adaptively tuned. To ensure robustness, each method is evaluated over five independent runs with different random seeds, with the mean and variance of each metric reported.

4.2. Performance Analysis

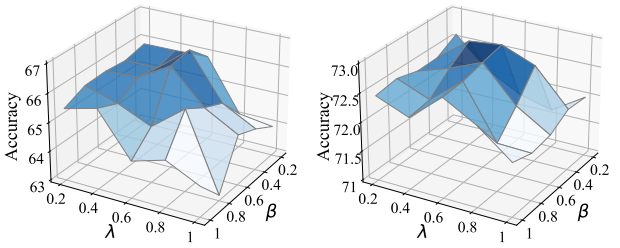
Table 1 and 2 report the best average performance of all methods across two real-world datasets (Results for Bail-Bs are in Appendix A.1). Several key observations can be drawn: (1) When comparing traditional learning methods with Fair GNNs, we observe that fairness baselines improve fairness performance at the expense of classification accuracy. (2) While fairness baselines aim to improve fairness, they struggle with classification under distribution shifts. In contrast, OOD methods achieve better classification but exhibit lower fairness across all test graphs due to their inability to learn fair representations. This highlights the need for fairness-aware modules to ensure fairness across varying distributions. (3) DANCE outperforms all other baselines in most cases, demonstrating the effectiveness of disentanglement in filtering out sensitive information and the adversarial module in handling distribution shifts. These results validate the importance of explicitly modeling both fairness constraints and distributional robustness to achieve a trade-off between fairness and classification performance.

4.3. Ablation Study

To assess the effectiveness of the various modules in DANCE and gain insight into their contributions, we compare DANCE against four variants (Extended ablation studies are provided in Appendix A.2): (1) Variant 1: It removes the Mixup module. (2) Variant 2: It removes the adversarial learning module. (3) Variant 3: It removes the group-acquired alignment module. (4) Variant 4: It removes the representation disentanglement module. The results of DANCE and its variants are presented in Table 3, with key observations as follows: (1) Compared to Variant 1, removing the Mixup module significantly affects performance, as extending an accurate classification boundary relies on properly synthesizing features for generated nodes. (2) The comparison with Variant 2 reveals that adversarial learning effectively mitigates distribution shifts, improving both fairness and classification performance. (3) The comparison with Variant 3 demonstrates that the group-acquired alignment module is essential for aligning adversarial learning representations with original representations, significantly enhancing fair representation learning. (4) The comparison with Variant 4 confirms that the representation disentanglement module effectively separates sensitive information from learned representations, thereby enhancing fairness.

Table 3. Ablation studies on the variants of DANCE.

Dataset	Metric	Var1	Var2	Var3	Var4	DANCE
C1	ACC↑	78.50±0.15	78.51±0.32	**79.52±0.21**	78.30±0.13	78.58±0.41
	ROC-AUC↑	71.69±0.01	71.55±0.01	72.01±0.01	71.48±0.06	**72.18±0.08**
	$\Delta_{DP}\downarrow$	4.11±0.71	3.54±0.69	6.33±1.43	2.71±0.90	**0.74±0.56**
	$\Delta_{EO}\downarrow$	1.15±0.28	0.72±0.00	1.30±1.11	**0.38±0.05**	0.70±0.49
C2	ACC↑	79.65±0.40	79.58±0.34	**80.58±0.01**	79.42±0.42	78.4±0.62
	ROC-AUC↑	**76.82±0.05**	76.71±0.04	76.79±0.01	76.30±0.35	75.5±0.21
	$\Delta_{DP}\downarrow$	3.08±0.57	4.54±4.02	8.63±0.24	4.30±2.05	**1.62±1.53**
	$\Delta_{EO}\downarrow$	2.66±0.38	2.03±0.77	4.10±0.05	1.99±0.21	**0.40±0.08**
C3	ACC↑	72.90±0.37	72.48±1.52	72.66±0.02	**73.06±0.13**	72.77±0.73
	ROC-AUC↑	75.51±0.07	**75.95±0.08**	75.91±0.01	75.71±0.03	75.63±0.12
	$\Delta_{DP}\downarrow$	2.33±0.29	0.50±0.18	**0.43±0.01**	1.44±0.18	1.38±1.18
	$\Delta_{EO}\downarrow$	1.91±2.16	2.43±4.51	3.94±0.03	2.03±3.30	**0.31±0.04**
C4	ACC↑	72.92±0.37	72.35±0.76	**73.02±0.02**	72.76±0.01	72.19±0.71
	ROC-AUC↑	77.65±0.03	77.52±0.01	77.42±0.01	77.24±0.09	**77.56±0.02**
	$\Delta_{DP}\downarrow$	**1.02±0.21**	1.37±0.46	1.71±0.06	2.55±1.05	1.24±1.25
	$\Delta_{EO}\downarrow$	2.09±0.31	1.29±0.26	1.38±0.19	1.24±0.13	**0.36±0.20**



(a) Bail-B1 (b) Bail-B2

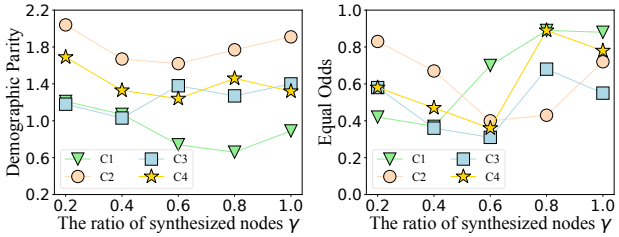
Figure 3. Comparison of performance w.r.t. different values of λ and β for the overall objective.

4.4. Parameter Analysis

We analyze the impact of three hyperparameter groups in DANCE: the loss weights λ as well as β in the objective function (Equation 16), the ratio γ of synthesized nodes, and the trade-off parameter δ in Mixup Strategy (Equation 7). Our key findings are as follows: (1) As depicted in Figure 3, excessively high values of λ and β negatively impact accuracy. The optimal values, approximately $\lambda = 0.6$ and $\beta = 0.4$, effectively balance accuracy and fairness. (2) As shown in Figure 4, the optimal ratio γ depends on the dataset and its distribution. However, an effective synthesized node ratio typically falls within the range of 0.5 to 0.6, leading to improved fairness performance. (3) When the trade-off parameter δ is fixed, Figure 5 shows that the optimal δ for both equal odds and demographic parity typically falls between 0.3 to 0.7. However, as no single value is optimal for all datasets, adaptive tuning is still required.

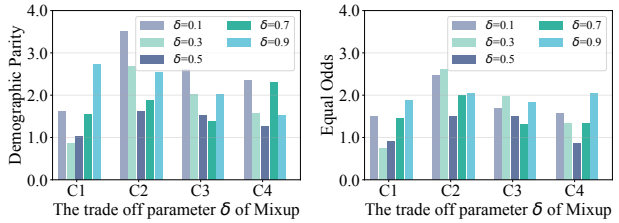
4.5. Visualization

We present an intuitive visualization of learned representations in both the training and test graphs on the Pokec dataset via t-SNE (Van der Maaten & Hinton, 2008). As



(a) DP (Credit) (b) EO (Credit)

Figure 4. Comparison of fairness w.r.t. different ratios γ for synthesized minor nodes.



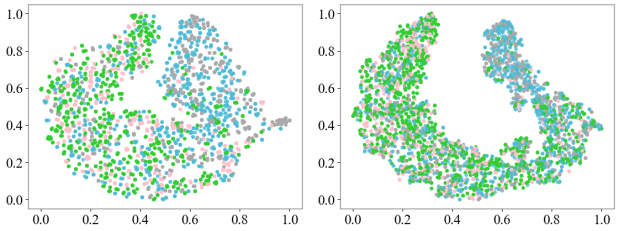
(a) DP (Credit) (b) EO (Credit)

Figure 5. Comparison of fairness w.r.t. different trade-off parameter δ for unbiased Mixup.

shown in Figure 6, nodes are color-coded by target and sensitive labels in the latent space. The representations of nodes within the same EO group remain closely aligned across training and test graphs, demonstrating that DANCE preserves representation consistency under distribution shifts. This suggests that DANCE mitigates data distribution shift, enhancing both fairness and classification generalization.

5. Related Work

Fairness in GNNs. Recent studies show that graph representation learning, particularly GNNs, tends to inherit biases from the training graph, with the message-passing mechanism in GNNs further amplifying these biases (Dai & Wang, 2021; Dong et al., 2023). Our work primarily addresses group fairness, which can be grouped into pre-, in-, and post-processing strategies according to the stage at which they are applied. Specifically, pre-processing strategies aim to modify the graph data to ensure fairness before the model is trained. EDITS (Dong et al., 2022) mitigate both graph attribute and structural bias with Wasserstein distance minimization between group pairs. In-processing strategies aim to adjust the learning process itself to promote fairness during model training. NIFTY (Agarwal et al., 2021) introduces a new multiple-objective function to enforce the fairness and stability of the GNN. Graphhair (Ling et al., 2023) employs automated augmentation to achieve fairness and informativeness in the generated graph. Post-processing strategies focus on adjusting the model's output to ensure fairness. FairGNN (Dai & Wang, 2021) incorporates an adversarial task for predictiong sensitive attributes, ensuring the GNN's predictions are independent of these attributes. However, these approaches all assume that the training and test graphs follow the same distribution.



(a) pokec_z (b) pokec_n

Figure 6. Visualizations of representations learned from the training and test graphs on the Pokec dataset.

Fairness under Distribution Shifts. Distribution shifts refer to situations where the test data distribution differs from that of the training data (Liu et al., 2021). Several studies have explored fairness under distribution shifts (Rezaei et al., 2021; Giguere et al., 2022; Lin et al., 2024). Shifty (An et al., 2022) reweights the training samples to match the group proportions present in the test samples. The algorithm (Mandal et al., 2020) treats test data as weighted combinations of training samples and focuses on ensuring fairness under the worst-case distribution shift. Additionally, some works (An et al., 2022) establish sufficient conditions for fairness transfer and propose self-training to balance and minimize consistency losses across different groups. Nevertheless, most of these methods are restricted to Euclidean data. Recently, FatraGNN (Li et al., 2024b) tackled fairness under distribution shifts in graphs. However, it overlooks the problem of group imbalance and unintentionally ignores task-related information during distribution shifts.

6. Conclusion

In this paper, we propose DANCE, a novel approach for graph fairness learning under distribution shifts. Unlike traditional methods that the training and test graphs share the same distribution, DANCE explicitly addresses graph distribution shifts commonly encountered in real-world scenarios. Specifically, we generate challenging yet unbiased virtual graph data in both graph and hidden spaces by incorporating unbiased mixup and fairness-aware adversarial learning to simulate distribution shifts from a data-centric view. Building on this, a group-acquired alignment objective and a representation disentanglement objective are further proposed to enhance the fairness in the model. Extensive experiments on several real-world datasets demonstrate that our DANCE effectively improves both fairness and model performance under distribution shifts, making it a promising solution for real-world graph learning tasks.

Impact Statement

The proposed DANCE framework advances fairness-aware graph learning under distribution shifts by integrating dual unbiased data expansion and group-acquired alignment. By generating challenging yet unbiased virtual graph data in both graph and feature spaces, DANCE simulates distribution shifts from a data-centric perspective. It leverages mixup-based minority group expansion and adversarial perturbation to enhance robustness, while explicitly disentangling sensitive and task-related information to improve fairness. Furthermore, the group-acquired alignment mechanism promotes consistency of representations across sensitive groups. This dual expansion and alignment strategy enables DANCE to achieve superior generalization and fairness on out-of-distribution graphs. The framework holds broad potential for real-world applications where demographic disparities and distribution shifts are prevalent, such as financial risk prediction, social network analysis, and criminal justice modeling.

Acknowledgement

This paper is partially supported by the National Key Research and Development Program of China with Grant No. 2023YFC3341203 and the National Natural Science Foundation of China (NSFC Grant Number 62276002 and 62306014) as well as the Fundamental Research Funds for the Central Universities in UIBE (Grant No. 23QN02).

References

Agarwal, C., Lakkaraju, H., and Zitnik, M. Towards a unified framework for fair and stable graph representation learning. In *Uncertainty in Artificial Intelligence*, pp. 2114–2124. PMLR, 2021.

An, B., Che, Z., Ding, M., and Huang, F. Transferring fairness under distribution shifts via fair consistency regularization. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 32582–32597, 2022.

Bengio, Y., Deleu, T., Rahaman, N., Ke, R., Lachapelle, S., Bilaniuk, O., Goyal, A., and Pal, C. A meta-transfer objective for learning to disentangle causal mechanisms. In *Proceedings of the International Conference on Learning Representations*, 2019.

Bose, A. and Hamilton, W. Compositional fairness constraints for graph embeddings. In *Proceedings of the International Conference on Machine Learning*, pp. 715–724, 2019.

Calders, T., Kamiran, F., and Pechenizkiy, M. Building classifiers with independency constraints. In *IEEE International Conference on Data Mining Workshops*, pp. 13–18, 2009.

Dai, E. and Wang, S. Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. In *Proceedings of the International ACM Conference on Web Search & Data Mining*, pp. 680–688, 2021.

Dai, E. and Wang, S. Learning fair graph neural networks with limited and private sensitive attribute information. *IEEE Transactions on Knowledge and Data Engineering*, 35(7):7103–7117, 2022.

Ding, F., Hardt, M., Miller, J., and Schmidt, L. Retiring adult: New datasets for fair machine learning. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 6478–6490, 2021.

Dong, Y., Liu, N., Jalaian, B., and Li, J. Edits: Modeling and mitigating data bias for graph neural networks. In *Proceedings of the Web Conference*, pp. 1259–1269, 2022.

Dong, Y., Ma, J., Wang, S., Chen, C., and Li, J. Fairness in graph mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10583–10602, 2023.

Gasteiger, J., Weißenberger, S., and Günnemann, S. Diffusion improves graph learning. In *Proceedings of the Conference on Neural Information Processing Systems*, 2019.

Giguere, S., Metevier, B., Brun, Y., Da Silva, B. C., Thomas, P. S., and Niekum, S. Fairness guarantees under demographic shift. In *Proceedings of the International Conference on Learning Representations*, 2022.

Guo, Z., Li, J., Xiao, T., Ma, Y., and Wang, S. Towards fair graph neural networks via graph counterfactual. In *Proceedings of the International Conference on Information and Knowledge Management*, pp. 669–678, 2023.

Han, X., Jiang, Z., Liu, N., and Hu, X. G-mixup: Graph data augmentation for graph classification. In *Proceedings of the International Conference on Machine Learning*, pp. 8230–8248, 2022.

Hardt, M., Price, E., and Srebro, N. Equality of opportunity in supervised learning. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 3323–33, 2016.

Jiang, Z., Han, X., Jin, H., Wang, G., Chen, R., Zou, N., and Hu, X. Chasing fairness under distribution shift: A model weight perturbation approach. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 63931–63944, 2023.

Jordan, K. L. and Freiburger, T. L. The effect of race/ethnicity on sentencing: Examining sentence type, jail length, and prison length. *Journal of Ethnicity in Criminal Justice*, 13(3):179–196, 2015.

Ju, W., Fang, Z., Gu, Y., Liu, Z., Long, Q., Qiao, Z., Qin, Y., Shen, J., Sun, F., Xiao, Z., et al. A comprehensive survey on deep graph representation learning. *Neural Networks*, pp. 106207, 2024.

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

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. In *Proceedings of the International Conference on Learning Representations*, 2017.

Köse, Ö. D. and Shen, Y. Fairness-aware node representation learning. *arXiv preprint arXiv:2106.05391*, 2021.

Li, F., Wang, B., Zhu, L., Li, J., Zhang, Z., and Chang, X. Cross-domain transfer hashing for efficient cross-modal retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024a.

Li, H., Wang, X., Zhang, Z., and Zhu, W. Out-of-distribution generalization on graphs: A survey. *arXiv preprint arXiv:2202.07987*, 2022a.

Li, H., Zhang, Z., Wang, X., and Zhu, W. Learning invariant graph representations for out-of-distribution generalization. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 11828–11841, 2022b.

Li, H., Zhang, Z., Wang, X., and Zhu, W. Invariant node representation learning under distribution shifts with multiple latent environments. *ACM Transactions on Information Systems*, 42(1):1–30, 2023a.

Li, P., Wang, Y., Zhao, H., Hong, P., and Liu, H. On dyadic fairness: Exploring and mitigating bias in graph connections. In *Proceedings of the International Conference on Learning Representations*, 2021.

Li, W.-Z., Wang, C.-D., Xiong, H., and Lai, J.-H. Graphsha: Synthesizing harder samples for class-imbalanced node classification. In *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1328–1340, 2023b.

Li, Y., Wang, X., Xing, Y., Fan, S., Wang, R., Liu, Y., and Shi, C. Graph fairness learning under distribution shifts. In *Proceedings of the Web Conference*, pp. 676–684, 2024b.

Li, Z., Dong, Y., Liu, Q., and Yu, J. X. Rethinking fair graph neural networks from re-balancing. In *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1736–1745, 2024c.

Liao, P., Zhao, H., Xu, K., Jaakkola, T., Gordon, G. J., Jegelka, S., and Salakhutdinov, R. Information obfuscation of graph neural networks. In *Proceedings of the International Conference on Machine Learning*, pp. 6600–6610, 2021.

Lin, Y., Zhao, C., Shao, M., Meng, B., Zhao, X., and Chen, H. Towards counterfactual fairness-aware domain generalization in changing environments. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 4560–4568, 2024.

Ling, H., Jiang, Z., Luo, Y., Ji, S., and Zou, N. Learning fair graph representations via automated data augmentations. In *Proceedings of the International Conference on Learning Representations*, 2023.

Liu, J., Shen, Z., He, Y., Zhang, X., Xu, R., Yu, H., and Cui, P. Towards out-of-distribution generalization: A survey. *arXiv preprint arXiv:2108.13624*, 2021.

Mandal, D., Deng, S., Jana, S., Wing, J., and Hsu, D. J. Ensuring fairness beyond the training data. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 18445–18456, 2020.

Meng, K., Wu, P., and Yang, X. Lipschitz continuity of solution multifunctions of extended ℓ_1 regularization problems. *arXiv preprint arXiv:2406.16053*, 2024.

Newman, M. E. Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582, 2006.

Pal, S. K. and Mitra, S. Multilayer perceptron, fuzzy sets, classifiaction. 1992.

Park, S. and Byun, H. Fair-vpt: Fair visual prompt tuning for image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12268–12278, 2024.

Park, S., Lee, J., Lee, P., Hwang, S., Kim, D., and Byun, H. Fair contrastive learning for facial attribute classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10389–10398, 2022.

Pham, T.-H., Zhang, X., and Zhang, P. Fairness and accuracy under domain generalization. In *Proceedings of the International Conference on Learning Representations*, 2023.

Ren, T., Zhang, H., Wang, Y., Ju, W., Liu, C., Meng, F., Yi, S., and Luo, X. Mhgc: Multi-scale hard sample mining for contrastive deep graph clustering. *Information Processing & Management*, 62(4):104084, 2025.

Rezaei, A., Liu, A., Memarrast, O., and Ziebart, B. D. Robust fairness under covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 9419–9427, 2021.

Spinelli, I., Scardapane, S., Hussain, A., and Uncini, A. Fairdrop: Biased edge dropout for enhancing fairness in graph representation learning. *IEEE Transactions on Artificial Intelligence*, 3(3):344–354, 2021.

Sui, Y., Wu, Q., Wu, J., Cui, Q., Li, L., Zhou, J., Wang, X., and He, X. Unleashing the power of graph data augmentation on covariate distribution shift. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 18109–18131, 2023.

Van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Volpi, R., Namkoong, H., Sener, O., Duchi, J. C., Murino, V., and Savarese, S. Generalizing to unseen domains via adversarial data augmentation. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 5339–5349, 2018.

Wang, P., Zhang, Z., Lei, Z., and Zhang, L. Sharpness-aware gradient matching for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3769–3778, 2023.

Wang, X., Liu, H., Shi, C., and Yang, C. Be confident! towards trustworthy graph neural networks via confidence calibration. In *Proceedings of the Conference on Neural Information Processing Systems*, pp. 23768–23779, 2021.

Wang, Y., Zhao, Y., Dong, Y., Chen, H., Li, J., and Derr, T. Improving fairness in graph neural networks via mitigating sensitive attribute leakage. In *Proceedings of the International ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1938–1948, 2022.

Wang, Z., Qiu, M., Chen, M., Salem, M. B., Yao, X., and Zhang, W. Toward fair graph neural networks via real counterfactual samples. *Knowledge and Information Systems*, 66(11):6617–6641, 2024.

Wu, Q., Zhang, H., Yan, J., and Wipf, D. Handling distribution shifts on graphs: An invariance perspective. *arXiv preprint arXiv:2202.02466*, 2022.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Philip, S. Y. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2020.

Yang, C., Wu, Q., Wang, J., and Yan, J. Graph neural networks are inherently good generalizers: Insights by bridging gnns and mlps. In *Proceedings of the International Conference on Learning Representations*, 2023.

Yeh, I.-C. and Lien, C.-h. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert systems with applications*, 36(2):2473–2480, 2009.

Yu, J., Liang, J., and He, R. Mind the label shift of augmentation-based graph ood generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11620–11630, 2023.

Zhang, F., Chen, C., Hua, X.-S., and Luo, X. Fate: Learning effective binary descriptors with group fairness. *IEEE Transactions on Image Processing*, 2024a.

Zhang, G., Yuan, G., Cheng, D., Liu, L., Li, J., and Zhang, S. Disentangled contrastive learning for fair graph representations. *Neural Networks*, 181:106781, 2025.

Zhang, H., Cissé, M., Dauphin, Y., and Lopez-Paz, D. mixup: Beyond empirical risk minimization.¡ i¿ iclr¡/i¿. In *Proceedings of the International Conference on Learning Representations*, 2018.

Zhang, K., Liu, S., Wang, S., Shi, W., Chen, C., Li, P., Li, S., Li, J., and Ding, K. A survey of deep graph learning under distribution shifts: from graph out-of-distribution generalization to adaptation. *arXiv preprint arXiv:2410.19265*, 2024b.

Zhuang, X., Zhang, Q., Wu, B., Ding, K., Fang, Y., and Chen, H. Graph sampling-based meta-learning for molecular property prediction. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 4729–4737, 2023.

Table 4. Performance comparison (%±σ) on Bail-Bs.

Dataset	Metric	MLP	GCN	FairVGNN	NIFTY	EDITS	EERM	CAF	SAGM	RFR	FatraGNN	DANCE
B1	ACC↑	70.53±1.01	72.93±4.06	69.76±2.03	69.54±7.26	72.69±1.72	73.25±1.4	69.39±2.30	73.08±4.25	71.63±1.52	86.64±0.93	**95.54±12.94**
	ROC-AUC↑	62.76±1.87	59.41±14.42	64.82±4.32	62.65±5.95	59.91±0.31	63.98±1.28	62.84±1.84	62.76±3.45	62.39±1.53	89.92±0.01	**99.75±0.00**
	$\Delta_{DP}\downarrow$	4.83±9.38	4.58±0.78	11.05±4.58	7.21±4.54	4.35±1.3	8.85±2.57	4.46±2.03	7.33±4.59	2.57±1.24	**3.45±2.87**	3.99±17.19
	$\Delta_{EO}\downarrow$	7.48±7.31	10.19±2.3	8.35±4.82	9.57±2.8	9.22±0.97	10.93±2.38	4.97±2.31	7.35±4.56	2.63±0.87	4.79±3.19	**1.41±0.15**
B2	ACC↑	64.33±0.63	69.88±0.45	65.03±2.4	69.95±8.3	69.03±0.16	70.2±0.12	69.36±1.37	68.67±3.24	68.62±1.42	90.48±0.44	**99.50±0.01**
	ROC-AUC↑	59.21±1.18	68.35±10.68	70.21±2.61	65.93±13.46	74.25±0.73	72.23±0.49	71.58±2.03	70.67±2.14	70.25±2.35	90.88±4.48	**99.81±0.00**
	$\Delta_{DP}\downarrow$	8.36±1.62	6.91±0.58	5.64±2.78	3.21±4.54	3.2±3.06	8.31±0.5	2.53±3.62	5.78±2.53	2.15±1.94	0.15±0.79	**0.07±0.00**
	$\Delta_{EO}\downarrow$	6.51±0.32	8.68±0.2	3.23±3.47	3.57±2.8	2.89±0.54	6.29±0.12	3.81±2.08	6.34±3.56	2.64±1.63	0.43±1.14	**0.26±0.12**
B3	ACC↑	60.76±0.18	68.56±4.2	70.63±0.61	68.8±9.76	68.56±1.82	70.69±5.42	68.97±2.44	69.50±2.12	68.36±1.89	91.57±4.65	**91.46±0.57**
	ROC-AUC↑	62.89±2.87	72.99±0.68	80.76±5.01	77.98±5.5	79.28±1.48	79.98±3.61	78.04±2.67	78.43±3.90	78.74±1.95	94.42±3.63	**99.06±0.13**
	$\Delta_{DP}\downarrow$	9.8±0.38	12.72±2.44	8.05±0.45	6.21±4.54	5.24±0.03	5.64±3.49	6.32±2.45	6.78±3.23	4.23±1.72	5.02±3.54	**2.86±0.06**
	$\Delta_{EO}\downarrow$	6.29±0.36	14.15±3.09	9.18±0.36	5.57±2.8	3.08±0.27	4.65±1.21	4.32±2.67	5.67±2.84	4.72±2.17	**2.43±4.94**	3.17±0.25
B4	ACC↑	63.13±1.69	69.43±0.48	68.99±2.44	57.96±11.99	68.42±0.14	70.9±1.36	67.33±2.67	70.88±0.98	69.18±2.68	90.95±3.39	**98.64±0.09**
	ROC-AUC↑	61.57±0.97	76.4±0.78	77.23±1.14	69.21±5.39	69.2±1.41	68.81±2.27	71.93±1.64	69.34±1.89	68.35±2.52	94.42±3.79	**99.71±0.01**
	$\Delta_{DP}\downarrow$	4.45±3.15	4.49±1.13	5.21±6.03	3.21±4.54	3.2±9.1	7.23±0.26	3.84±1.41	6.36±6.32	3.43±2.45	2.48±3.09	**0.41±0.04**
	$\Delta_{EO}\downarrow$	3.29±3.54	8.74±1.62	5.33±6.18	2.57±2.8	5.6±7.86	9.04±0.86	5.36±2.19	7.34±4.67	3.51±2.39	2.45±6.67	**0.75±0.01**
	Rank	11	10	6	9	3	8	4	7	5	2	**1**

A. Additional Results

A.1. Results on Bail-Bs

Table 4 shows classification and fairness performance on Bail-Bs.

A.2. Extended Ablation Study on Mixup and the Adversarial Module

To evaluate the effectiveness of the various modules in DANCE and understand their functions, we compare DANCE with six variants: (1) Variant 1: It selects anchor nodes through random sampling rather than hard sampling. (2) Variant 2: It removes the mixup module. (3) Variant 3: It configures the adversarial function to maximize only the classification component, excluding the fairness term. (4) Variant 4: It configures the adversarial function to maximize only the fairness component, excluding the classification term. (5) Variant 5: It eliminates the group-acquired alignment module. (6) Variant 6: It eliminates the representation disentanglement module.

The results of DANCE and its variants are presented in Table 3. Our key observations are as follows: (1) Impact of hard sampling and mixup. Compared to Variant 1, random sampling alone has a limited effect on final performance. Instead, constructing sufficiently challenging samples is essential for improving model performance. Compared to Variant 2, removing the mixup module has a more significant impact on final performance, as extending an accurate classification boundary relies on appropriately synthesizing features for the generated nodes. (2) Effect of adversarial learning components. The comparison with Variant 3 and Variant 4 reveals that the max component of the adversarial function enhances generalization to out-of-distribution data for target classification, whereas the min component focuses on fairness optimization in such settings. Consequently, Variant 3 achieves better performance in target label classification, while Variant 4 excels in fairness representation. (3) Role of group-acquired alignment. The comparison with Variant 5 demonstrates that the contrastive alignment module plays a crucial role in aligning adversarial learning representations with the original representations. This alignment significantly enhances fair representation learning. (4) Effectiveness of representation disentanglement. The comparison with Variant 6 confirms that the disentanglement contrast module effectively separates sensitive information from the representation learning process, thereby facilitating the learning of fairer representations.

B. Theorem Proof

Theorem 3.1: For any $i \in \mathcal{V}_s$, there exists $j \in \mathcal{V}_{-s}$ such that

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} \neq 0;$$

Table 5. Extended ablation studies on the variants of DANCE.

	metric	Var1	Var2	Var3	Var4	Var5	Var6	DANCE
C1	ACC↑	78.21±0.31	78.01±0.12	78.90±0.19	78.51±0.32	**79.52±0.21**	78.30±0.13	78.58±0.41
	ROC-AUC↑	71.15±0.03	71.46±0.02	71.57±0.01	71.55±0.01	72.01±0.01	71.48±0.06	**72.18±0.08**
	$\Delta_{DP}\downarrow$	1.21±0.43	2.88±0.67	3.55±2.88	3.54±0.69	6.33±1.43	2.71±0.90	**0.74±0.56**
	$\Delta_{EO}\downarrow$	0.52±0.12	0.74±0.30	0.49±0.05	0.72±0.00	1.30±1.11	**0.38±0.05**	0.70±0.49
C2	ACC↑	79.21±0.31	79.33±0.39	80.02±0.02	79.58±0.34	**80.58±0.01**	79.42±0.42	78.4±0.62
	ROC-AUC↑	75.9±0.28	74.6±0.11	76.74±0.02	76.71±0.04	**76.79±0.01**	76.30±0.35	75.5±0.21
	$\Delta_{DP}\downarrow$	2.24±0.57	3.11±0.73	5.76±0.73	4.54±4.02	8.63±0.24	4.30±2.05	**1.62±1.53**
	$\Delta_{EO}\downarrow$	0.87±0.62	1.87±0.33	2.49±0.39	2.03±0.77	4.10±0.05	1.99±0.21	**0.40±0.08**
C3	ACC↑	72.79±0.42	72.62±0.10	72.06±0.79	72.48±1.52	72.66±0.02	**73.06±0.13**	72.77±0.73
	ROC-AUC↑	75.28±0.20	75.30±0.12	75.86±0.02	**75.95±0.08**	75.91±0.01	75.71±0.03	75.63±0.12
	$\Delta_{DP}\downarrow$	1.18±0.07	2.18±0.15	1.00±0.58	0.50±0.18	**0.43±0.01**	1.44±0.18	1.38±1.18
	$\Delta_{EO}\downarrow$	0.81±0.47	0.91±0.10	2.48±3.62	2.43±4.51	3.94±0.03	2.03±3.30	**0.31±0.04**
C4	ACC↑	72.21±0.37	72.33±0.57	72.78±0.11	72.35±0.76	**73.02±0.02**	72.76±0.01	72.19±0.71
	ROC-AUC↑	77.01±0.01	77.03±0.03	77.51±0.04	77.52±0.01	77.42±0.01	77.24±0.09	**77.56±0.02**
	$\Delta_{DP}\downarrow$	1.69±0.17	1.31±0.29	**0.90±0.07**	1.37±0.46	1.71±0.06	2.55±1.05	1.24±1.25
	$\Delta_{EO}\downarrow$	0.55±0.35	1.35±0.19	0.95±0.14	1.29±0.26	1.38±0.19	1.24±0.13	**0.36±0.20**

For any $i \in \mathcal{V}_{-s}$, for all $j \in \mathcal{V}_s$, we have

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} = 0.$$

Remark: Theorem One indicates that the graph diffusion method can precisely control the information propagation between different groups, which is crucial for improving the fairness and performance of the model when dealing with data from different groups.

Proof: First, we consider the propagation within and out of the minority group. Let $i \in \mathcal{V}_s$. According to matrix multiplication, we have

$$[\boldsymbol{T}_{anc}^0]_{ij} = \delta_{ij},$$

$$[\boldsymbol{T}_{anc}^1]_{ij} = [\boldsymbol{A}_{anc}\boldsymbol{D}_{anc}^{-1}]_{ij} = \boldsymbol{A}_{anc,ij}/\boldsymbol{D}_{anc,jj}.$$

Since when constructing $\boldsymbol{A}_{anc}$, for $i \in \mathcal{V}_s$ exists $j \notin \mathcal{V}_s$ such that $\boldsymbol{A}_{anc,ij} \neq 0$, and $\sum_{r=0}^{\infty} |\theta^r|$ converges. We have

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} = \sum_{r=0}^{\infty} \theta^r \sum_{k=1}^{|\mathcal{V}|} [\boldsymbol{T}_{anc}^r]_{ik} [\boldsymbol{H}^{(l-1)}]_{kj}.$$

Hence, for a specific $j \notin \mathcal{V}_s$, $\sum_{r=0}^{\infty} \theta^r \sum_{k=1}^{|\mathcal{V}|} [\boldsymbol{T}_{anc}^r]_{ik} [\boldsymbol{H}^{(l-1)}]_{kj} \neq 0$, which means that information from $i \in \mathcal{V}_s$ can be propagated to nodes $j \notin \mathcal{V}_s$.

Then, we show how this method blocks the propagation to the non-minority group. Let $i \in \mathcal{V}_{-s}$, for all $j \in \mathcal{V}_s$, $\boldsymbol{A}_{anc,ij} = 0$. Suppose that for all $j \in \mathcal{V}_s$ and $s = 1, \ldots, r$,

$$[\boldsymbol{T}_{anc}^s]_{ij} = 0.$$

We have

$$[\boldsymbol{T}_{anc}^{r+1}]_{ij} = \sum_{k=1}^{|\mathcal{V}|} [\boldsymbol{T}_{anc}^r]_{ik} [\boldsymbol{T}_{anc}^1]_{kj}.$$

Since $[\boldsymbol{T}_{anc}^1]_{kj} = 0$ (for $k, j \in \mathcal{V}_s$) and $[\boldsymbol{T}_{anc}^r]_{ik} = 0$ (for $k \in \mathcal{V}_s$), we have $[\boldsymbol{T}_{anc}^{r+1}]_{ij} = 0$ (for $j \in \mathcal{V}_s$). Therefore,

$$\left[\sum_{r=0}^{\infty} \theta^r (\boldsymbol{T}_{anc}^r \boldsymbol{H}^{(l-1)})\right]_{ij} = 0 \quad (j \in \mathcal{V}_s).$$

Theorem 3.2: Suppose that the loss function $\mathcal{F}(\omega)$ is differentiable with respect to the parameter ω, and there exists a constant $M > 0$ such that the gradient of the loss function is M-Lipschitz continuous. We adopt the SGD algorithm to update the parameter. When an appropriate learning rate α is chosen such that $0 < \alpha < \frac{2}{M}$, we have $\lim_{t\to\infty} \|\nabla\mathcal{F}(\omega_t)\| = 0$ and $\lim_{t\to\infty} \mathcal{F}(\omega_t)$ exists, which means the loss function $\mathcal{F}(\omega)$ converges.

Remark: The Lipschitz continuity assumption limits the rate of change of the gradient of the loss function, which is a common condition for the convergence proof of many optimization algorithms.

Proof: Based on the M-Lipschitz continuity, we have

$$\mathcal{F}(\omega_2) \leq \mathcal{F}(\omega_1) + \nabla\mathcal{F}(\omega_1)^T(\omega_2 - \omega_1) + \frac{M}{2}\|\omega_2 - \omega_1\|^2.$$

Let $\omega_2 = \omega_{t+1}$ and $\omega_1 = \omega_t$, then we have

$$\mathcal{F}(\omega_{t+1}) \leq \mathcal{F}(\omega_t) - \alpha(1 - \frac{M\alpha}{2})\|\nabla\mathcal{F}(\omega_t)\|^2.$$

When $0 < \alpha < \frac{2}{M}$, we have $\mathcal{F}(\omega_{t+1}) \leq \mathcal{F}(\omega_t)$. Moreover, since $\mathcal{F}(\omega)$ has a lower bound, according to the monotone convergence theorem, $\lim_{t\to\infty} \mathcal{F}(\omega_t)$ exists.

Next, we prove that $\lim_{t\to\infty} \|\nabla\mathcal{F}(\omega_t)\| = 0$. From the analysis above, we can get

$$\alpha(1 - \frac{M\alpha}{2})\|\nabla\mathcal{F}(\omega_t)\|^2 \leq \mathcal{F}(\omega_t) - \mathcal{F}(\omega_{t+1}).$$

Sum both sides of the inequality from $t = 0$ to $T - 1$:

$$\alpha(1 - \frac{M\alpha}{2})\sum_{t=0}^{T-1} \|\nabla\mathcal{F}(\omega_t)\|^2 \leq \mathcal{F}(\omega_0) - \mathcal{F}(\omega_T).$$

Since $\lim_{t\to\infty} \mathcal{F}(\omega_t)$ exists, $\sum_{t=0}^{\infty} \|\nabla\mathcal{F}(\omega_t)\|^2$ converges. According to the necessary condition for the convergence of a series, we have $\lim_{t\to\infty} \|\nabla\mathcal{F}(\omega_t)\| = 0$.