Fair Text-Attributed Graph Representation Learning

Ruilin Luo[1], **Tianle Gu**[1], **Lin Wang**[2], **Yunfeng Zhou**[3], **Songtao Jiang**[4], **Lei Wang**[5], **Yujiu Yang**[1†*]
[1]Tsinghua University, [2]The Chinese University of Hong Kong (Shenzhen)
[3]Beijing University of Technology [4]Zhejiang University
[5]Ping An Technology (Shenzhen) Co., Ltd.

Abstract

Text-Attributed Graphs (TAGs), which integrate text and graph structures, have recently gained traction, especially in web applications. However, as a graph structure, TAG representation learning (TAGRL) naturally inherits issues from Graph Neural Networks (GNNs), such as fairness. Moreover, previous TAGRL research has mainly focused on using LM-as-encoder to boost downstream task performance, with little consideration given to whether this process may raise additional concerns related to fairness and other safety-related issues. As the first work to explore fairness in TAGRL, this paper proposes the concept of evolving LM-as-encoder to LM-as-fair-encoder, developing a two-stage fairness-aware alignment process called FairTAG based on the observed issues. Specifically, we first mitigate the tendency of LMs to overfit to homophily during downstream tasks fine-tuning, followed by subgraph-level connection behavior preference optimization for selected anchor nodes. We provide theoretical support and demonstrate the feasibility of LM-as-fair-encoder through extensive experiments and ablation studies. We also show that FairTAG can be seamlessly integrated with fairness-enhancing strategies on the GNNs decoder side, thus innovatively constructing a plug-and-play learning framework.

1 Introduction

Text-Attributed Graphs (TAGs) are a prevalent data structure where nodes are enriched with textual information, combining structured and textual knowledge (Pan et al., 2024; Huang et al., 2024; Zhao et al., 2024; Jiang et al., 2024). Widely used in applications like recommendation systems and information retrieval (Wei et al., 2024; He et al., 2024b; Tang et al., 2024c; Zhang et al., 2024a), these graphs are increasingly central to decision-making systems. However, like their traditional counterparts, TAGs inherit significant fairness challenges from Graph Neural Networks (GNNs) (Chu et al., 2024; Dong et al., 2023; Li et al., 2024; Wang et al., 2022). Ensuring that models trained on TAGs do not perpetuate or amplify biases across demographic groups is a critical, yet underexplored, problem.

The dominant paradigm for Text-Attributed Graph Representation Learning (TAGRL) is *LM-as-encoder* (He et al., 2023; Jin et al., 2024; Duan et al., 2023), where a Language Model (LM) is fine-tuned to generate rich node embeddings. As shown in Figure 1, this paradigm has successfully improved task performance (e.g., AUC). However, our preliminary analysis, using the well-established Mixed-dyadic Demographic Parity (DP_m) metric (Spinelli et al., 2021), reveals a troubling trend: state-of-the-art LM-as-encoder methods often exhibit fairness scores equal to or worse than simpler methods, particularly on social network data. This raises a crucial question: are LMs, despite their power, inadvertently amplifying biases present in the graph?

The core intuition, which underpins our work, is that this fairness degradation is a direct consequence of how LMs learn on biased graph data. Real-world graphs are often dominated by **homophily**, where intra-group links vastly outnumber inter-group links. When fine-tuned, LMs tend to overfit to this majority pattern. They can achieve low training loss by simply learning this "easy" homophilous pattern, while failing to correctly model the "harder," less frequent, but crucial inter-group connections. This creates a significant fairness problem.

To address this, we propose advancing the paradigm from LM-as-encoder to **LM-as-fair-encoder** and introduce **FairTAG**, a two-stage framework designed to achieve both effectiveness and fairness. Our approach is built on a clear logical progression from a coarse-grained global cor-

* Email: yang.yujiu@sz.tsinghua.edu.cn. † Corresponding author.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 14330–14353
November 4-9, 2025

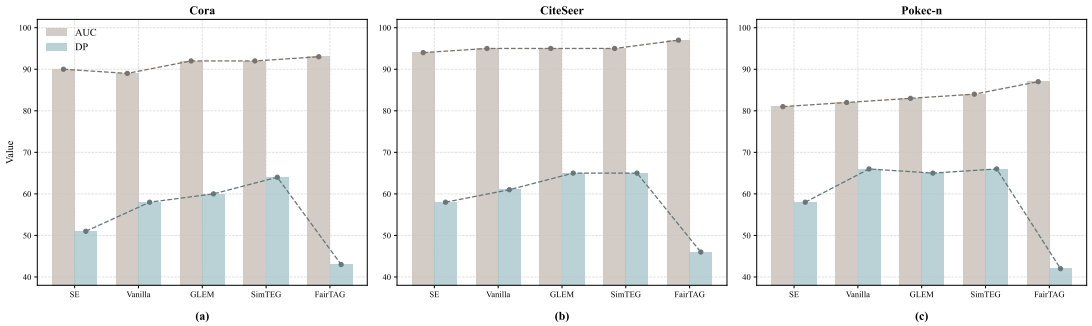

Figure 1: Figures (a)-(c) illustrate that while advanced LM-as-encoder methods (GLEM, SimTEG) improve AUC over baselines, they often worsen the fairness metric DP_m. Figure (d) shows the inherent imbalance in real-world graphs, where intra-group positive links are far more common than inter-group ones.

rection to a targeted, local refinement.

Stage 1: Global Debiasing with Fairness Score-based Edge Drop (FSED). Our first step is to combat the model's tendency to overfit to "easy" samples. We define *simple samples* as edges a model predicts correctly with high confidence (overwhelmingly the majority-class, intra-group links) and *difficult samples* as those it misclassifies or predicts with low confidence (disproportionately the minority-class, inter-group links). To systematically identify these, we first train an *oracle model* on the original biased data to serve as a stable proxy for the difficulty inherent in the dataset. FSED then uses the discrepancy between the oracle's predictions and the ground truth to assign a difficulty score to each edge. By probabilistically dropping the simplest, high-confidence homophilous edges, FSED effectively re-balances the training data's difficulty. This forces the subsequent model to learn from a more challenging dataset, compelling it to grasp the nuanced patterns of inter-group connections rather than relying on the easy homophily shortcut.

Stage 2: Local Refinement with Anchor Node Preference Optimization (ANPO). While FSED provides a crucial global correction, it treats all nodes monolithically. However, a node deep within a homophilous cluster faces different challenges than one at the boundary between groups. To address this limitation, we introduce a targeted, local intervention. We identify *anchor nodes* as nodes at the boundaries of sensitive attribute clusters, which have a more balanced mix of intra- and inter-group connections and are most vulnerable to fairness violations. For these critical nodes, we use ANPO, a preference optimization algorithm inspired by DPO (Rafailov et al., 2024), to explicitly teach the model to value inter-group connections more equitably. We frame this as a preference task: for a given anchor node, a valid inter-group connection is *preferred* over a valid intra-group connection. This allows us to surgically target the local fairness gap with a level of precision that a global strategy cannot achieve.

Our main contributions are summarized as follows:

- We are the first to systematically investigate and address the fairness problem in modern LM-based TAGRL, proposing a novel two-stage framework, FairTAG, that evolves the LM-as-encoder paradigm.
- We provide a clear, motivated methodology that combines a global, difficulty-aware data resampling strategy (FSED) with a targeted, local preference optimization technique (ANPO) for surgical bias correction.
- We provide theoretical analysis showing how LM biases can be amplified by GNNs and how our proposed methods mitigate this effect.
- Extensive experiments demonstrate that FairTAG significantly improves fairness across multiple metrics while maintaining or improving link prediction performance. Our framework is also plug-and-play, compatible with existing fairness-enhancing GNN decoders.

2 Related Work

LM Training on TAGs. Early approaches for node features on Text-Attributed Graphs (TAGs) utilized shallow embeddings like Skip-gram (Mikolov et al., 2013) or BoW (Harris,

1954). With the rise of Pretrained Language Models (PLMs), methods emerged to enhance representations by fine-tuning LMs on graph-related tasks. These fine-tuned embeddings then serve as input to GNNs for structural learning, as seen in TextGNN (Zhu et al., 2021), GIANT (Chien et al., 2021), and SimTEG (Duan et al., 2023). Self-supervised learning (Fang et al., 2024) and iterative training schemes like GLEM (Zhao et al., 2022) and DRAGON (Chien et al., 2021) further integrate text and structure. The LLM era has spurred works using them for richer node/subgraph descriptions (Tang et al., 2024b) and downstream tasks via prompting or collaborative inference (Sun et al., 2023; He et al., 2024a). Efforts to build graph foundation models are also prominent recently (Ye et al., 2023; Tang et al., 2024a; Shi et al., 2024).

Fairness on Graphs. Graph fairness is a growing concern, although fairness on TAGs remains less explored. Prior work often employed graph augmentation or adversarial learning for GNN fairness (Hussain et al., 2022; Zhang et al., 2023a; Li et al., 2021; Singer and Radinsky, 2022; Zhang et al., 2023b, 2024b; Liu et al., 2023). For instance, adversarial methods (Liao et al., 2021; Zhang et al., 2023a) optimize node representations. FairWalk (Rahman et al., 2019) adjusts random walk probabilities based on sensitive attributes. FairDrop (Spinelli et al., 2021), a graph augmentation method, balances edge proportions between nodes with different/same sensitive attributes to manage privacy information flow (Liu et al., 2023). DropEdge (Rong et al., 2019) randomly removes edges to combat overfitting/oversmoothing. FairGT (Kose and Shen, 2024) adapted graph transformers for fairness. Graph fairness research extends to applications like recommendation systems (Wu et al., 2021; Fu et al., 2020; Chen et al., 2024). LM-driven graph fairness is a promising direction in the LLM era.

3 Preliminary

Text-Attributed Graphs. A Text-Attributed Graph (TAG) is $\mathcal{G} = (\mathcal{V}, A, \mathbf{X})$, with node set $\mathcal{V}$ ($|\mathcal{V}| = N$), adjacency matrix $A \in \mathbb{R}^{N\times N}$ (defining edge set $\mathcal{E}$ where $A_{pq} = 1 \implies (v_p, v_q) \in \mathcal{E}$), and text attributes $\mathbf{X} \in \mathbb{R}^{N\times d}$.

Fair Graph Augmentation. Fair graph augmentation aims to neutralize information transfer biases related to sensitive attributes $\mathcal{S} = \{s_i | v_i \in \mathcal{V}\}$ (Yang et al., 2024). The edge set $\mathcal{E}$ is divided into $\mathcal{E}_{inter}$ (connecting nodes with different sensitive attributes) and $\mathcal{E}_{intra}$ (connecting nodes with the same sensitive attributes):

$$\mathcal{E}_{inter} = \{(v_i, v_j) | s_i \neq s_j, \forall v_i, v_j \in \mathcal{V}\}. \quad (1)$$

A masking matrix $\mathcal{M}$ identifies inter-group connections:

$$m_{ij} = \begin{cases} 1, & (v_i, v_j) \in \mathcal{E}_{inter}, \forall v_i, v_j \in \mathcal{V} \\ 0, & otherwise \end{cases} \quad (2)$$

$\mathcal{M}$ is used in two main ways. First, for edge dropping, FairDrop (Spinelli et al., 2021) uses a random mechanism controlled by $\delta \in [0, \frac{1}{2}]$:

$$rr(m_{ij}) = \begin{cases} m_{ij}, & \text{with probability: } \frac{1}{2} + \delta \\ 1 - m_{ij}, & \text{with probability: } \frac{1}{2} - \delta \end{cases} \forall v_i, v_j \in \mathcal{V} \quad (3)$$

to create an adjusted adjacency matrix $\tilde{A} = A \circ rr(M)$, where larger δ retains more $\mathcal{E}_{inter}$ edges. Second, methods like FairSIN (Yang et al., 2024) modify edge weights in A to enhance inter-group information flow while minimizing loss.

LMs Fitting on TAGs. For an LM $\mathcal{F}(\cdot)$, vanilla methods directly embed node text. Supervised/self-supervised fine-tuning of LMs on graph-based objectives or downstream tasks yields more domain-specific embeddings. For link prediction, a typical loss is:

$$\mathcal{L} = \text{CrossEntropy}(\phi(\mathcal{F}(t_{src}), \mathcal{F}(t_{dst})), Y) \quad (4)$$

where ϕ is a prediction head (e.g., MLP) and Y is the edge label.

4 Methodology

In this section, we introduce our fairness-aware LM training paradigm, FairTAG, which is the first framework to focus on fair LM training on TAGs. This pipeline can be seamlessly integrated with existing GNN architectures and connected with fairness research in the GNN domain, demonstrating its versatility. We propose a two-stage alignment method to mitigate unfairness at the embedding level before applying GNNs. This includes: i) a fairness-score based edge drop strategy that reduces the learning inertia of LMs on both inter-class and intra-class edges; and ii) a subgraph-level preference optimization to enhance local fairness in predictions. Newly defined symbols, as well as those from Section 3, are used in this section. A detailed summary of the notation is provided in Appendix H.

Theorem 4.1. *When considering a K-layer GNN, if the following assumptions hold: For any node pair $s_i = s_j$, the features processed by language model $\mathcal{F}(\cdot)$ result in a reduced distance between nodes with the same sensitive attribute, that is:*

$$\mathbb{E}[\| x_i^{LM} - x_j^{LM} \|^2] \leq \mathbb{E}\| x_i^{(0)} - x_j^{(0)} \|^2 - \delta \quad (5)$$

$\delta > 0$ as the contraction amount, then the within-class and between-class differences in link prediction probabilities increase with the number of layers K and the dominant eigenvalue λ_1, thereby worsening the DP.

$$\Delta_{DP}^{LM} \geq \Delta_{DP}^{(0)} + C \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k} \quad (6)$$

The proof is in Appendix B.1. Additionally, we conduct an analysis to measure the average squared Euclidean distance between intra-class node pairs after different encoding stages. The results in Table 1 shows a clear reduction in distance, strenthening our assumption.

Table 1: A comparison of different embedding methods on various datasets. The values in parentheses indicate the percentage change relative to the Shallow Embedding baseline.

Dataset	Shallow Embedding	Vanilla LM	SimTEG
Cora	26.3	20.9 (-20.5%)	12.5 (-52.5%)
CiteSeer	53.9	45.5 (-15.6%)	34.8 (-35.5%)
PubMed	2.11	1.87 (-11.4%)	0.99 (-53.1%)
Pokec-n	44.4	20.1 (-54.7%)	15.1 (-66.0%)

4.1 Fairness Score Based Edge Drop

Unlike the fairness-related work at the GNN end, on a TAG, the text modality is a crucial source of information. Therefore, using random edge masking would result in the loss of important information when training LMs. We first propose Fairness Score-based Edge Drop (FSED) strategy. Our first step is to train an **oracle model** $\mathcal{F}_{oracle}$ on the training set without any preprocessing of the training data. We can align the LMs with the downstream tasks using full fine-tuning or Parameter-Efficient Fine-Tuning (PEFT) methods. As illustrated by the concerns shown in Figure 1, we regard the oracle model as a system fitted over all edges, thereby providing parameter-level guidance for edge selection. We judge which edges are truly valuable based on the difference between the model's decision confidence and the ground-truth labels, thereby correcting the training data. Finally, we train another LM on the debiased data.

Oracle Model Training Given the full training set $\mathcal{E}$, given $i \in [0, |\mathcal{E}|)$, the finetune process is formulated as follows:

$$P(e_i) = \phi(\mathcal{F}_{oracle}(t_{i\_src}), \mathcal{F}_{oracle}(t_{i\_dst})) \quad (7)$$

$$\mathcal{L} = \sum_{i=1}^{|\mathcal{E}|} \text{CrossEntropy}(P(e_i), y_i) \quad (8)$$

Fairness-aware Edge Drop The oracle model $\mathcal{F}_{oracle}$ fine-tuned on the training set is trained on a biased edge set; even if the model has relatively completely aligned with the linking information on the graph, it also inherits the learning from a biased perspective. Specifically, due to the homophily assumption, real-world datasets often exhibit more positive samples in $\mathcal{E}_{intra}$ ($|\mathcal{E}_{intra}^{+}| > |\mathcal{E}_{intra}^{-}|$) and more negative samples in $\mathcal{E}_{inter}$ ($|\mathcal{E}_{inter}^{-}| > |\mathcal{E}_{inter}^{+}|$). The LM's sensitivity to text enables it to more accurately learn biases that make it easier to judge links. However, this may not necessarily meet fairness requirements. We have $E = \{e_i\}_{i=1}^{|\mathcal{E}|}$ and propose using the model's output logits $P(E)$ as the LM's confidence in judging the positivity or negativity of samples, and define the absolute distance between them and the ground-truth labels Y as the value score for whether they can be selected as training samples.

$$Q(E) = \text{normalize}(|P(E) - Y|) \quad (9)$$

$$\text{Score}(E) = Q(E) + \gamma \cdot (1 - Q(E)) \quad (10)$$

$$\text{Sel}(i) = \begin{cases} 1, & \text{if uniform}(0,1) < \text{Score}(e_i), \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

$1 \leq i \leq |\mathcal{E}|$. The hyperparameter γ is used to scale the scores, thereby controlling the intensity of the contrast in the data. When γ is small, the data distribution pays more attention to the difficult samples, that is, $\mathcal{E}_{intra}^{-}$ and $\mathcal{E}_{inter}^{+}$. The Sel vector divides $\mathcal{E}$ into the selected samples $\mathcal{E}_{sel}$ and the unselected samples $\mathcal{E}_{un}$.

Fairness-aware Training In this step, we fine-tune another LM (annotated as $\mathcal{F}_{ref}$) on the selected $\mathcal{E}_{sel}$. In this process, we encourage the model to achieve a more balanced training on $\mathcal{E}_{sel}$, while we also hope that the model maintains its original judgment on the simple samples in $\mathcal{E}_{un}$. Therefore, we perform knowledge distillation on

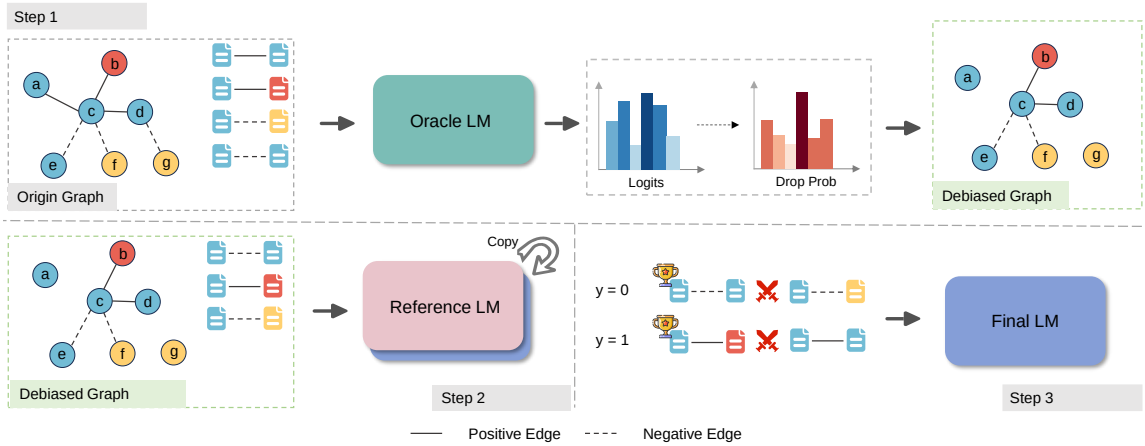

Figure 2: The overall pipeline of proposed FairTAG. In the first step, an oracle model is used to fine-tune on the link prediction task and output fairness scores for edge dropping. For example, the edges (a, c) and (d, g) in the diagram may likely be dropped due to their high scores for correctly predicted positive edges from $\mathcal{E}^+$ and low scores for correctly predicted negative edges from $\mathcal{E}^-$. In the second step, a reference language model (ref LM) is fine-tuned on the edges selected after the first step. In the third step, the final language model (final LM) generates preferences for inter-relation and intra-relation based on the anchor node.

the $\mathcal{E}_{un}$ set from the Oracle model side, allowing the new model to maintain the judgment of the Oracle model as much as possible on $\mathcal{E}_{un}$. The two-point distribution for the labels derived from Equation 7 is denoted as $\psi(y|e)$. The total training loss for $\mathcal{F}_{ref}(\cdot)$ is formulated as below:

$$\mathcal{L}_1 = \sum_{i=1}^{|\mathcal{E}_{sel}|} \text{CrossEntropy}(P_{ref}(e_i), y_i) + \sum_{j=1}^{|\mathcal{E}_{un}|} \mathbb{D}_{KL}(\psi_{ref}(y|e_j) \parallel \psi_{oracle}(y|e_j)) \tag{12}$$

Theorem 4.2. *FSED corrects the covariance structure of LM feature distribution to suppress intra-class over-clustering. The lower bound in Theorem 4.1 corrects as follows:*

$$\Delta_{DP}^{LM_{sel}} \leq \Delta_{DP}^{LM} - C \cdot \alpha \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k} \tag{13}$$

where $\alpha > 0$ is variance recovery coefficient introduced by FSED.

The proof is in Appendix B.2.

4.2 Anchor Node Preference Optimization

While the model $\mathcal{F}_{ref}$ is trained on debiased data, its optimization remains tied to the global graph structure. We propose LMs can achieve finer-grained locality fairness using anchor nodes. These nodes, typically at the edge of sensitive information clusters with a balanced ratio of intra- and inter-class connections, are intended to dilute the representational distance between difficult and simple samples.

$$\mathcal{V}^{\pm} = \{v_i \mid v_i \in \mathcal{V}, \exists v_j \in \mathcal{V}, (v_i, v_j) \in \mathcal{E}^{\pm}_{\text{intra}}, \wedge \exists v_k \in \mathcal{V}, (v_i, v_k) \in \mathcal{E}^{\pm}_{\text{inter}}\} \tag{14}$$

As Equation 14 indicates, a node can be in both $\mathcal{V}^+$ and $\mathcal{V}^-$. Our goal is to optimize the probability gap for different connections of these anchor nodes. For instance, for nodes in $\mathcal{V}^+$, we aim to raise the lower bound of logits for inter-relations being positive, thus narrowing the gap with intra-relations and mitigating unfairness. Inspired by DPO (Rafailov et al., 2024; Lai et al., 2024), we employ a preference optimization algorithm.

We frame link prediction for anchor nodes as an *edge generation* process. In text generation, preference alignment rewards one answer over another for a given prompt x. For debiased link prediction, however, when tasked to '*predict which relation is more likely to be positive*', the model learns that identifying harder samples yields higher reward. Crucially, we avoid optimizing by suppressing the log-likelihood of the less preferred side (intra-relations) (Pal et al., 2024), as the model's tendency to score intra-relations highly is not inherently an empirical error. Let $\psi(e|y) = Normalize(P(e))$ be the normalized probability distribution of edges locally extended from anchor

Table 2: Results comparison on four benchmarks. EM represents the Embedding Method (EM). We report AUC for performance metrics, and report ΔDP_m, ΔEO_m, ΔDP_g, ΔEO_g, ΔDP_s and ΔEO_s for fairness evaluation. The best and second-best performances are highlighted in blue and red.

EM	GCN							GAT						
	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
Link Prediction on Cora.														
SE	90.4±0.6	51.3±2.6	18.8±3.9	18.3±3.8	22.1±5.6	89.1±5.6	100.0±0.0	88.0±0.4	45.9±1.8	18.8±2.7	14.9±3.1	19.2±3.1	81.9±3.7	100.0±0.0
Vanilla	88.1±0.8	53.4±1.5	26.0±4.6	17.5±3.1	21.1±4.1	88.1±5.2	100.0±0.0	85.0±0.9	50.8±1.7	28.4±3.5	16.1±3.2	18.7±4.5	86.1±7.0	100.0±0.0
GLEM	92.8±1.0	55.9±1.8	24.9±4.1	19.2±3.3	25.1±3.9	92.1±5.0	100.0±0.0	89.5±0.6	49.1±1.9	22.5±3.2	17.2±2.9	20.3±3.5	85.2±4.6	100.0±0.0
SimTEG	92.4±0.6	52.2±1.8	21.2±4.6	15.2±3.4	19.2±3.1	89.6±3.0	100.0±0.0	90.3±0.5	47.6±2.3	20.4±3.3	13.1±2.7	16.6±1.7	82.7±4.3	100.0±0.0
FairTAG	93.9±0.6	43.6±2.5	13.1±4.5	11.4±3.9	11.3±4.7	73.8±3.2	100.0±0.0	91.7±0.8	42.8±3.2	15.6±5.2	11.6±3.1	16.8±5.1	78.1±3.7	100.0±0.0
Link Prediction on CiteSeer.														
SE	93.7±0.5	57.5±1.7	39.0±3.6	20.7±4.1	13.1±3.8	80.4±2.1	72.5±12.6	93.9±0.4	55.4±1.5	28.7±4.2	20.6±3.3	14.2±2.4	78.1±2.7	67.8±6.0
Vanilla	94.7±0.4	57.7±1.7	22.5±2.8	23.3±3.2	12.6±2.2	83.6±2.6	66.0±10.6	94.2±0.5	57.5±1.7	24.7±2.2	22.8±3.4	12.5±2.8	82.2±3.3	66.2±10.4
GLEM	95.1±0.5	58.9±1.5	23.3±3.8	20.5±2.9	14.7±2.6	84.5±3.2	70.3±14.6	93.7±0.5	55.9±1.7	25.5±3.1	21.1±2.8	15.5±2.3	81.3±3.5	68.8±12.3
SimTEG	95.2±0.6	57.5±1.3	13.1±2.4	16.6±2.7	12.7±2.3	76.3±5.9	62.7±15.8	95.2±0.6	52.2±1.5	12.6±2.6	18.1±2.8	13.8±2.9	75.8±5.2	60.6±15.4
FairTAG	96.3±0.6	42.7±1.5	8.7±1.7	13.1±3.2	9.7±2.0	68.6±4.8	51.4±18.4	95.5±0.4	46.1±1.2	10.1±1.6	13.8±2.2	12.3±3.4	71.0±3.1	57.2±13.4
Link Prediction on PubMed.														
SE	94.3±0.2	46.1±0.7	17.0±1.8	5.6±1.2	4.9±0.8	60.5±2.0	34.8±5.8	90.6±0.2	44.1±0.8	20.9±1.2	3.9±1.0	5.2±0.8	59.9±1.9	38.7±3.9
Vanilla	94.5±0.5	40.1±1.1	14.0±0.8	6.5±0.5	5.4±0.8	38.5±2.3	21.1±2.1	86.0±0.2	43.0±0.6	16.8±1.1	6.0±1.1	6.5±0.7	28.3±0.8	16.1±3.1
GLEM	95.2±0.4	41.5±1.2	15.5±1.1	6.8±0.9	5.7±0.8	40.5±2.1	20.5±2.5	87.2±0.5	40.1±0.9	17.1±1.2	6.1±0.8	6.1±0.7	30.5±1.2	17.5±2.3
SimTEG	95.2±0.3	39.8±1.8	16.1±1.3	7.5±0.9	6.0±1.1	30.8±1.0	18.4±3.6	88.3±0.5	37.1±0.7	15.2±1.0	6.1±0.6	5.3±0.8	27.9±1.1	17.9±3.2
FairTAG	96.0±0.4	29.1±1.1	9.4±0.8	2.2±0.6	3.2±0.4	21.1±2.0	13.9±2.0	88.8±0.7	27.4±1.2	10.1±0.6	2.5±0.9	3.3±0.7	20.9±1.8	12.3±1.3
Link Prediction on Pokec-n.														
SE	81.2±0.5	58.1±1.2	28.5±2.3	14.6±1.9	9.1±1.2	14.6±1.9	38.5±3.5	80.9±0.6	56.8±1.5	29.1±2.1	13.5±1.7	8.9±1.5	15.1±2.3	39.1±4.1
Vanilla	82.5±0.6	60.9±1.5	30.1±2.1	16.9±1.7	11.5±1.5	16.9±1.7	41.9±4.1	81.2±0.7	58.1±1.2	31.5±2.3	15.6±1.9	10.1±1.2	17.6±1.9	42.5±3.5
GLEM	83.1±0.7	59.5±1.3	29.8±2.5	15.8±1.8	10.9±1.3	15.8±1.8	40.8±3.8	83.6±0.8	57.5±1.3	30.1±2.1	14.9±1.7	9.5±1.5	16.9±1.7	41.9±4.1
SimTEG	83.9±0.8	62.1±1.7	29.9±2.1	17.9±1.2	11.8±1.2	16.6±1.7	39.1±3.1	83.1±0.9	56.1±1.2	29.5±2.3	13.6±1.9	8.1±1.2	15.6±1.9	40.5±3.5
FairTAG	84.9±0.9	47.5±1.3	18.7±1.6	8.5±1.0	4.6±1.1	8.5±1.1	28.1±2.3	84.2±0.8	48.1±1.2	21.5±1.9	7.1±1.1	3.9±1.6	9.0±1.0	31.5±2.8

nodes. We then propose the **A**nchor **N**ode-based **P**reference **O**ptimization (ANPO) loss as follows*:

$$\mathcal{L}_{pos} = -\sum_{v\in\mathcal{V}^+} \mathbb{E}_{e_w\sim\mathcal{E}^+_{v,inter}, e_l\sim\mathcal{E}^+_{v,intra}} \left[\log\sigma\left(\beta\left(\log\frac{\psi_\theta(e_w|y)}{\psi_{ref}(e_w|y)} - \beta\log\frac{\psi_\theta(e_l|y)}{\psi_{ref}(e_l|y)} - \lambda\max\left(0, \log\frac{\psi_{ref}(e_w|y)}{\psi_\theta(e_w|y)}\right)\right)\right)\right] \tag{15}$$

$$\mathcal{L}_{neg} = \sum_{v\in\mathcal{V}^-} \mathbb{E}_{e_w\sim\mathcal{E}^-_{v,intra}, e_l\sim\mathcal{E}^-_{v,inter}} \left[\log\sigma\left(\beta\left(\log\frac{\psi_\theta(e_w|y)}{\psi_{ref}(e_w|y)} - \beta\log\frac{\psi_\theta(e_l|y)}{\psi_{ref}(e_l|y)} + \lambda\max\left(0, \log\frac{\psi_\theta(e_w|y)}{\psi_{ref}(e_w|y)}\right)\right)\right)\right] \tag{16}$$

$$\mathcal{L}_2 = \mathcal{L}_{pos} + \mathcal{L}_{neg} \tag{17}$$

Theorem 4.3. *Equation 17 can effectively model the edge generation pattern of anchor nodes through preference optimization.*

The derivation is placed in Appendix B.3.

Theorem 4.4. *After ANPO, node representations can compress the inter-class mean differences, thereby alleviating unfairness.*

The proof is in Appendix B.4.

Given Theorem 4.3 and 4.4, we have theoretical support to demonstrate the superiority of ANPO.

4.3 Integrated with GNN Decoder

After obtaining the debiased $\mathcal{F}_{final}$, we integrated it with GNN decoder to perform link prediction, the loss is demonstrated in Equation 19.

$$h_v^{(k)} = COMBINE(h_v^{k-1}, AGG(h_u^{k-1} : u \in \mathcal{N}_v)) \tag{18}$$

$$\mathcal{L}_{lp} = \sum_{i=1}^{|\mathcal{E}|} CE(\phi(h_{i\_src}, h_{i\_dst}), y_i) \tag{19}$$

*More details in Appendix G.

Table 3: Performance Comparison of FairTAG and GNN decoder side methods.

Method	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)
Link Prediction on Cora					
FairDrop	90.7±0.7	49.5±2.8	19.0±3.5	17.5±3.5	21.5±5.0
DropEdge	90.9±0.6	50.1±2.5	18.2±4.0	18.5±3.9	20.9±5.2
FairGT	92.5±0.7	45.0±2.8	15.5±4.0	13.0±3.5	14.5±4.5
G-FAME++	93.8±0.8	44.1±3.6	15.8±3.7	12.9±2.8	7.5±2.3
FairTAG	93.9±0.6	43.6±2.5	13.1±4.5	11.4±3.9	11.3±4.7
Link Prediction on CiteSeer					
FairDrop	94.0±0.5	55.0±1.8	39.5±3.5	19.8±4.0	12.5±3.6
DropEdge	94.1±0.4	56.5±1.6	38.0±3.8	20.9±4.2	13.5±3.9
FairGT	95.0±0.5	45.0±1.6	15.0±3.0	14.5±3.5	10.5±3.0
G-FAME++	91.9±0.1	38.6±0.5	13.0±1.7	13.6±0.2	8.5±1.1
FairTAG	96.3±0.6	42.7±1.5	8.7±1.7	13.1±3.2	9.7±2.0
Link Prediction on PubMed					
FairDrop	94.6±0.3	35.0±0.8	17.2±1.7	5.2±1.1	4.5±0.7
DropEdge	94.7±0.2	35.5±0.6	16.5±1.9	5.7±1.3	5.0±0.9
FairGT	95.3±0.3	33.0±1.0	10.5±1.0	3.0±0.8	3.5±0.5
G-FAME++	95.6±0.1	35.9±0.0	11.0±0.6	2.3±0.2	1.5±0.1
FairTAG	96.0±0.4	29.1±1.1	9.4±0.8	2.2±0.6	3.2±0.4
Link Prediction on Pokec-n					
FairDrop	81.5±0.6	57.0±1.3	28.8±2.2	14.0±1.8	8.8±1.1
DropEdge	81.7±0.5	57.5±1.1	28.0±2.4	14.7±2.0	9.2±1.3
FairGT	83.0±0.8	50.0±1.4	21.0±2.0	10.0±1.5	6.0±1.2
G-FAME++	83.5±1.0	50.2±1.5	20.4±1.8	10.1±1.2	6.3±1.1
FairTAG	84.9±0.9	47.5±1.3	18.7±1.6	8.5±1.0	4.6±1.1

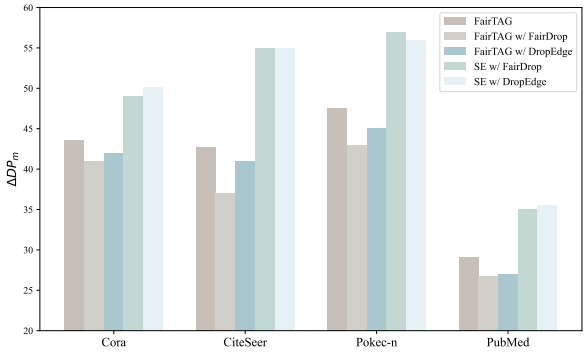

Figure 3: We employ FairDrop and DropEdge as fairness-aware training methods on the decoder side, demonstrating the compatibility of FairTAG in constructing cascaded fair training.

5 Experiment

5.1 Experimental Setup

Dataset We use five popular dataset in fair graph learning, i.e., *Cora*, *CiteSeer*, *PubMed*, *Pokec-n* and *Pokec-z*. The detailed information of five used benchmarks are demonstrated in Appendix C.1. Due to the similarity of Pokec-n and Pokec-z, we report related results of Pokec-n in the main paper, and place results on Pokec-z in Appendix.

Evaluation Metrics For link prediction performance, we use AUC as a metric. For fairness metrics, we use *Demographic Parity* (DP) (Dwork et al., 2012) and *Equalized Odds* (EO) (Hardt et al., 2016) as evaluation metrics. Their designs in the field of graph theory have been expanded by (Spinelli et al., 2021; Masrour et al., 2020) into tests under different dyadic groups, with detailed introductions provided in the Appendix E.

Compared Methods and Backbone We aim to demonstrate that our method, in comparison with existing LM encoding techniques, can provide GNNs with more fair initial embeddings enhanced by text information. For the baselines, we opt for Shallow Embedding (Grover and Leskovec, 2016), Vanilla (direct encoding by a LM), GLEM (Zhao et al., 2022) and SimTEG (Duan et al., 2023) for comparison. Regarding the fairness training methods on the GNN side, we select FairDrop (Spinelli et al., 2021), DropEdge (Rong et al., 2019), FairGT (Luo et al., 2024) and G-FAME++ (Liu et al., 2023). And we choose GCN (Kipf and Welling, 2016) and GAT (Veličković et al., 2017) as our GNN backbones.

Implementation Details We report the mean and standard deviation of ten runs with random seeds 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. The experiments are implemented using PyTorch and run on NVIDIA A100 GPUs. We employ *all-roberta-large-v1* (Reimers, 2019; Liu, 2019) as the language model (LM) and use full-parameter fine-tuning. The parameter search is detailed in Appendix F.

5.2 Results Comparison

Comprehensive experiments on four datasets demonstrate our method's superiority in AUC and fairness metrics.

Does our method provide more effective and fair representations? Using GCN as the backbone, we observe that other LM-based approaches often maintain or even exacerbate unfairness, particularly on the Pokec-n dataset. We speculate this occurs because citation network texts, despite thematic content, include method-specific statements preventing over-clustering. Conversely, Pokec dataset texts, comprising only personal information, lead to overly clustered features for individuals with similar characteristics. In contrast, FairTAG excels in fairness metrics, achieving a comprehensive lead. Specifically, for Metric ΔDP_m, FairTAG outperforms the runner-up by 11.0%, 15.4%, 26.8%, and 18.4% on the four datasets, respectively. For AUC, FairTAG also outperformed other baselines, except on PubMed, where all methods performed relatively poorly.

How does performance compare with fairness-aware GNN methods? To further evaluate FairTAG's effectiveness, we compare it with recent fairness-aware training methods on the decoder side. FairTAG demonstrates a clear advantage on Cora, PubMed, and Pokec-n, effectively optimizing both AUC and fairness. This suggests that in text-rich scenarios, optimizing embedding approaches may be more efficient.

Can our method demonstrate generalization across different GNN backbones? Results in Table 2 shows that GAT yields poorer AUC compared to GCN. However, FairTAG still shows an advantage in AUC and fairness over other LM-based approaches when using a GAT backbone. This suggests FairTAG's adaptability to different GNN backbones. As illustrated in Table 3, FairTAG achieves an absolute advantage on Cora, PubMed, and Pokec-n, effectively optimizing both AUC and fairness, reinforcing that optimizing embedding approaches can be highly efficient in text-rich graph.

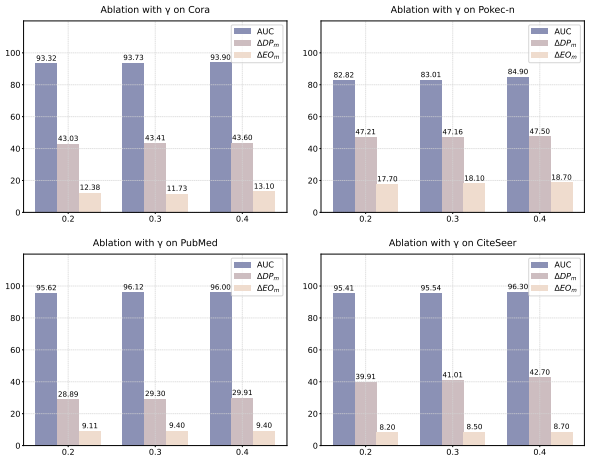

Figure 4: Ablation study on γ in FSED. We use AUC, ΔDP_m and ΔEO_m to demonstrate the performance of accuracy and fairness.

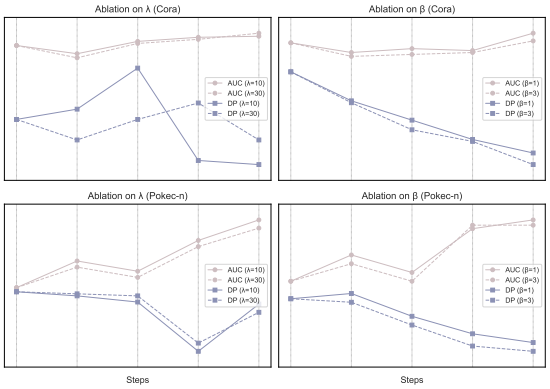

Figure 5: We conduct ablation studies on two key parameters of ANPO. When experimenting with parameter λ, we keep $\beta = 10$. When experimenting with parameter β, we keep $\lambda = 0.3$.

Table 4: Comparison of node classification on Pokec-n and Pokec-z datasets.

Method	Pokec-n			Pokec-z		
	Acc ↑	DP ↓	EO ↓	Acc ↑	DP ↓	EO ↓
SE	68.6 ± 0.5	3.8 ± 0.9	2.9 ± 1.2	66.8 ± 1.1	4.0 ± 1.0	2.8 ± 1.0
Vanilla	69.0 ± 0.4	3.9 ± 0.6	3.0 ± 1.0	67.3 ± 1.0	4.5 ± 1.3	2.9 ± 0.8
SimTEG	69.1 ± 0.5	4.0 ± 0.8	3.2 ± 1.0	68.2 ± 1.6	4.6 ± 1.0	4.3 ± 1.7
GLEM	67.8 ± 0.8	3.6 ± 0.7	3.4 ± 0.9	68.0 ± 0.8	3.9 ± 1.0	2.3 ± 1.7
FairVGNN	64.9 ± 1.2	1.7 ± 0.8	1.8 ± 0.7	67.3 ± 1.7	1.8 ± 1.2	1.3 ± 1.0
FairSIN	67.9 ± 0.3	0.6 ± 0.2	0.4 ± 0.4	69.2 ± 0.3	1.5 ± 0.7	0.6 ± 0.5
FairGAT	67.1 ± 0.4	2.6 ± 0.5	1.6 ± 0.9	68.2 ± 1.1	0.7 ± 0.7	1.2 ± 0.6
FairTAG	69.5 ± 0.7	1.2 ± 0.2	0.6 ± 0.3	69.9 ± 1.0	1.4 ± 0.6	0.5 ± 0.4

6 Compatibility with GNN-side Methods

To bridge the gap with fairness-aware methods on the GNN decoder side and establish FairTAG as a plug-and-play approach, we demonstrate its compatibility. Specifically, FairTAG, when trained with GNNs using FairDrop or DropEdge, exhibits superior performance for ΔDP_m (Figure 3). This indicates FairTAG, pioneering fairness optimization on the LM side, is compatible with decoder-side techniques, enabling a cascaded optimization pipeline. We also show that individual use of FairDrop or DropEdge does not match the performance of joint training. This establishes FairTAG as an effective plug-and-play method in TAGRL.

7 Ablation Studies

7.1 Ablation on FSED

Figure 4 presents the results of the ablation study on the parameter γ in the first stage alignment. Firstly, the selection of the γ involves a certain trade-off. For example, in the overall trend, when γ increases and the distribution approaches the original distribution, FSED tends to capture the original features more, thereby improving the AUC, but at the expense of fairness. However, the difference in the overall magnitude of change is acceptable, because the improvement compared to other methods is actually more widespread.

7.2 Ablation on Parameters of ANPO

In this section, we conduct ablation studies on key factors in the ANPO process to explore the roles of β and λ in different objectives. When analyzing one parameter, we keep the other constant and take checkpoints during training to examine the trajectory of changes for various objectives. The specific results are shown in Figure 4, and our key observations are as follows: i) β is more critical for fairness metrics. This observation aligns with our fundamental objective, as our preference optimization is inherently designed to influence edge selection behavior preferences rather than directly fitting the labels. However, the magnitude of the parameter does not directly indicate superiority or inferiority. On the Cora and Pokec-n datasets, increasing β yields different relative results. ii) Similar to the results in DPOP, λ can be used for stable improvement in accuracy. Specifically, in the ablation experiments on β, the AUC exhibits varying degrees of fluctuation as training progresses. However, this fluctuation is much more stable in the two line charts on the left, quickly converging towards the optimal value.

8 Cross Task Generalization

To demonstrate generalizability across different tasks, we employed a two-layer MLP as the connec-

tion head for the node classification task on the LM trained with FairTAG. The results in Table 4 show performance and fairness that are on par with some leading fairness-aware methods. This suggests that for embedding strategies on the LM side, the focus might be more on the training approach rather than the specific targeted tasks.

9 Conclusion

Our work pioneers fairness in LM and GNN collaboration on TAGs. We identify fairness concerns in LM-as-encoder architectures and demonstrate that LM-specific techniques can mitigate bias during feature embedding. Our proposed two-stage alignment method, FairTAG, offers an innovative plug-and-play approach compatible with existing fair GNN training methods. FairTAG uniquely incorporates an edge drop based on the fairness score and an anchor node-based edge generation, enhancing fairness at both the global and subgraph levels. Empirically, FairTAG surpasses current LM-based and leading GNN decoder side techniques in link prediction accuracy and fairness. Theoretically, we highlight that the LM-induced bias can be amplified by GNNs; our method mitigates this by reducing the LM encoding bias and correcting the lower bound of the amplified bias. This research underscores the potential of re-envisioning LMs for fairness, paving the way for more equitable AI systems on text-rich graph data.

10 Limitations

Our work focuses primarily on the series of LM-as-encoder approaches. However, we have not yet considered the fairness risks associated with the recent graph foundation models. As the first work to propose fairness-aware training and theoretical support in text-rich graph scenarios, we can further extend this as future work. In addition, our work has been validated mainly on citation networks and social networks. In the future, we can explore rich text graphs in other scenarios, such as medicine-related fields.

11 Ethical Consideration

Our research is dedicated to advancing fairness in TAGRL. The proposed FairTAG framework aims to mitigate biases that can be inherited or amplified by GNNs and LMs, with a focus on improving demographic parity and equalized odds in link prediction tasks. We acknowledge that "fairness" is a complex, context-dependent concept. The metrics used, while standard, may not capture all fairness dimensions relevant to every application. Furthermore, while FairTAG aims to reduce unfairness propagation, it operates on existing data, which may itself contain inherent biases. Therefore, the method is a step towards fairer models but not a complete solution to underlying data-level biases. Responsible deployment of FairTAG requires careful consideration of the specific application context, ongoing monitoring, and an understanding that fairness-enhancing algorithms are tools to aid, not replace, human oversight and ethical judgment in decision-making processes. Besides, AI-assistant writing is employed in this paper for polishing.

12 Acknowledgements

This work was partly supported by the National Natural Science Foundation of China (Grant No. 62576191) and the research grant No. CT20240905126002 of the Doubao Large Model Fund.

References

Chen Cai. 2023. *Local-to-global perspectives on graph neural networks*. University of California, San Diego.

Wei Chen, Yiqing Wu, Zhao Zhang, Fuzhen Zhuang, Zhongshi He, Ruobing Xie, and Feng Xia. 2024. Fairgap: Fairness-aware recommendation via generating counterfactual graph. *ACM Transactions on Information Systems*, 42(4):1–25.

Eli Chien, Wei-Cheng Chang, Cho-Jui Hsieh, Hsiang-Fu Yu, Jiong Zhang, Olgica Milenkovic, and Inderjit S Dhillon. 2021. Node feature extraction by self-supervised multi-scale neighborhood prediction. *arXiv preprint arXiv:2111.00064*.

Zhibo Chu, Zichong Wang, and Wenbin Zhang. 2024. Fairness in large language models: A taxonomic survey. *ACM SIGKDD Explorations Newsletter*, 26(1):34–48.

Yushun Dong, Jing Ma, Song Wang, Chen Chen, and Jundong Li. 2023. Fairness in graph mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10583–10602.

Keyu Duan, Qian Liu, Tat-Seng Chua, Shuicheng Yan, Wei Tsang Ooi, Qizhe Xie, and Junxian He. 2023. Simteg: A frustratingly simple approach improves textual graph learning. *arXiv preprint arXiv:2308.02565*.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226.

Yi Fang, Dongzhe Fan, Daochen Zha, and Qiaoyu Tan. 2024. Gaugllm: Improving graph contrastive learning for text-attributed graphs with large language models. *arXiv preprint arXiv:2406.11945*.

Zuohui Fu, Yikun Xian, Ruoyuan Gao, Jieyu Zhao, Qiaoying Huang, Yingqiang Ge, Shuyuan Xu, Shijie Geng, Chirag Shah, Yongfeng Zhang, et al. 2020. Fairness-aware explainable recommendation over knowledge graphs. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 69–78.

Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864.

Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. 2023. Harnessing explanations: Llm-to-lm interpreter for enhanced text-attributed graph representation learning. *arXiv preprint arXiv:2305.19523*.

Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. 2024a. Harnessing explanations: LLM-to-LM interpreter for enhanced text-attributed graph representation learning. In *The Twelfth International Conference on Learning Representations*.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024b. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *arXiv preprint arXiv:2402.07630*.

Xuanwen Huang, Kaiqiao Han, Yang Yang, Dezheng Bao, Quanjin Tao, Ziwei Chai, and Qi Zhu. 2024. Gnns as adapters for llms on text-attributed graphs. In *The Web Conference 2024*.

Hussain Hussain, Meng Cao, Sandipan Sikdar, Denis Helic, Elisabeth Lex, Markus Strohmaier, and Roman Kern. 2022. Adversarial inter-group link injection degrades the fairness of graph neural networks. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 975–980. IEEE.

Songtao Jiang, Tuo Zheng, Yan Zhang, Yeying Jin, Li Yuan, and Zuozhu Liu. 2024. Med-moe: Mixture of domain-specific experts for lightweight medical vision-language models. *Preprint*, arXiv:2404.10237.

Bowen Jin, Gang Liu, Chi Han, Meng Jiang, Heng Ji, and Jiawei Han. 2024. Large language models on graphs: A comprehensive survey. *IEEE Transactions on Knowledge and Data Engineering*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

O Deniz Kose and Yanning Shen. 2024. Fairgat: Fairness-aware graph attention networks. *ACM Transactions on Knowledge Discovery from Data*, 18(7):1–20.

Xin Lai, Zhuotao Tian, Yukang Chen, Senqiao Yang, Xiangru Peng, and Jiaya Jia. 2024. Step-dpo: Step-wise preference optimization for long-chain reasoning of llms. *arXiv preprint arXiv:2406.18629*.

Peizhao Li, Yifei Wang, Han Zhao, Pengyu Hong, and Hongfu Liu. 2021. On dyadic fairness: Exploring and mitigating bias in graph connections. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Yibo Li, Xiao Wang, Yujie Xing, Shaohua Fan, Ruijia Wang, Yaoqi Liu, and Chuan Shi. 2024. Graph fairness learning under distribution shifts. In *Proceedings of the ACM on Web Conference 2024*, pages 676–684.

Peiyuan Liao, Han Zhao, Keyulu Xu, Tommi Jaakkola, Geoffrey J Gordon, Stefanie Jegelka, and Ruslan Salakhutdinov. 2021. Information obfuscation of graph neural networks. In *International conference on machine learning*, pages 6600–6610. PMLR.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Zheyuan Liu, Chunhui Zhang, Yijun Tian, Erchi Zhang, Chao Huang, Yanfang Ye, and Chuxu Zhang. 2023. Fair graph representation learning via diverse mixture-of-experts. In *Proceedings of the ACM Web Conference 2023*, pages 28–38.

Renqiang Luo, Huafei Huang, Shuo Yu, Xiuzhen Zhang, and Feng Xia. 2024. Fairgt: A fairness-aware graph transformer. *arXiv preprint arXiv:2404.17169*.

Farzan Masrour, Tyler Wilson, Heng Yan, Pang-Ning Tan, and Abdol Esfahanian. 2020. Bursting the filter bubble: Fairness-aware network link prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 841–848.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. 2024. Smaug: Fixing failure modes of preference optimisation with dpo-positive. *arXiv preprint arXiv:2402.13228*.

Bo Pan, Zheng Zhang, Yifei Zhang, Yuntong Hu, and Liang Zhao. 2024. Distilling large language models for text-attributed graph learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 1836–1845.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Tahleen A. Rahman, Bartlomiej Surma, Michael Backes, and Yang Zhang. 2019. Fairwalk: Towards fair graph embedding. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3289–3295. ijcai.org.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. 2019. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*.

Chuan Shi, Junze Chen, Jiawei Liu, and Cheng Yang. 2024. Graph foundation model. *Frontiers of Computer Science*, 18(6):186355.

Uriel Singer and Kira Radinsky. 2022. Eqgnn: Equalized node opportunity in graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 8333–8341.

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

Shengyin Sun, Yuxiang Ren, Chen Ma, and Xuecang Zhang. 2023. Large language models as topological structure enhancers for text-attributed graphs. *arXiv preprint arXiv:2311.14324*.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024a. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 491–500.

Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Long Xia, Dawei Yin, and Chao Huang. 2024b. Higpt: Heterogeneous graph language model. *arXiv preprint arXiv:2402.16024*.

Yanran Tang, Ruihong Qiu, Yilun Liu, Xue Li, and Zi Huang. 2024c. Casegnn: Graph neural networks for legal case retrieval with text-attributed graphs. In *European Conference on Information Retrieval*, pages 80–95. Springer.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Ruijia Wang, Xiao Wang, Chuan Shi, and Le Song. 2022. Uncovering the structural fairness in graph contrastive learning. *Advances in neural information processing systems*, 35:32465–32473.

Yifei Wang, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. 2021. Dissecting the diffusion process in linear graph convolutional networks. *Advances in Neural Information Processing Systems*, 34:5758–5769.

Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Llmrec: Large language models with graph augmentation for recommendation. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 806–815.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. 2019. Simplifying graph convolutional networks. In *International conference on machine learning*, pages 6861–6871. Pmlr.

Le Wu, Lei Chen, Pengyang Shao, Richang Hong, Xiting Wang, and Meng Wang. 2021. Learning fair representations for recommendation: A graph-based perspective. In *Proceedings of the Web Conference 2021*, pages 2198–2208.

Cheng Yang, Jixi Liu, Yunhe Yan, and Chuan Shi. 2024. Fairsin: Achieving fairness in graph neural networks through sensitive information neutralization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9241–9249.

Ruosong Ye, Caiqi Zhang, Runhui Wang, Shuyuan Xu, Yongfeng Zhang, et al. 2023. Natural language is all a graph needs. *arXiv preprint arXiv:2308.07134*, 4(5):7.

Binchi Zhang, Yushun Dong, Chen Chen, Yada Zhu, Minnan Luo, and Jundong Li. 2023a. Adversarial attacks on fairness of graph neural networks. *arXiv preprint arXiv:2310.13822*.

He Zhang, Xingliang Yuan, Quoc Viet Hung Nguyen, and Shirui Pan. 2023b. On the interaction between node fairness and edge privacy in graph neural networks. *arXiv preprint arXiv:2301.12951*.

Mengmei Zhang, Mingwei Sun, Peng Wang, Shen Fan, Yanhu Mo, Xiaoxiao Xu, Hong Liu, Cheng Yang, and Chuan Shi. 2024a. Graphtranslator: Aligning graph model to large language model for open-ended tasks. In *Proceedings of the ACM on Web Conference 2024*, pages 1003–1014.

Zhongjian Zhang, Mengmei Zhang, Yue Yu, Cheng Yang, Jiawei Liu, and Chuan Shi. 2024b. Endowing pre-trained graph models with provable fairness. In *Proceedings of the ACM on Web Conference 2024*, pages 1045–1056.

Huanjing Zhao, Beining Yang, Yukuo Cen, Junyu Ren, Chenhui Zhang, Yuxiao Dong, Evgeny Kharlamov, Shu Zhao, and Jie Tang. 2024. Pre-training and prompting for few-shot node classification on text-attributed graphs. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4467–4478.

Jianan Zhao, Meng Qu, Chaozhuo Li, Hao Yan, Qian Liu, Rui Li, Xing Xie, and Jian Tang. 2022. Learning on large-scale text-attributed graphs via variational inference. *arXiv preprint arXiv:2210.14709*.

Jason Zhu, Yanling Cui, Yuming Liu, Hao Sun, Xue Li, Markus Pelger, Tianqi Yang, Liangjie Zhang, Ruofei Zhang, and Huasha Zhao. 2021. Textgnn: Improving text encoder via graph neural network in sponsored search. In *Proceedings of the Web Conference 2021*, pages 2848–2857.

Appendices Content

A Graph Neural Networks

In this section, we introduce the used GNN backbones. GNNs are highly effective in processing graph data information, and their goal is to update node representation using message passing of the neighbor information. Specifically,

$$h_v^{(k)} = COMBINE(h_v^{k-1}, AGG(h_u^{k-1} : u \in \mathcal{N}_v)) \tag{20}$$

where $h_v^{(k)}$ denotes the representation of node v in the k-th layer and $\mathcal{N}_v = \{u|A_{v,u} = 1\}$ is a node set has a directed edge to v, in which A represents the adjacency matrix of the graph $\mathcal{G}$. AGG is a function used to aggregate information from neighboring nodes, and $COMBINE$ is a function to update node representation with aggregated information. The initial representation $h_v^{(0)}$ of a node v may come from some surface features, or be generated with the help of a LM.

B Theorems and Proofs

B.1 Bias Amplification

Theorem B.1. *When considering a K-layer GNN, if the following assumptions hold: For any node pair (i, j) with $s_i = s_j$, the features processed by language model $\mathcal{F}(\cdot)$ result in a reduced distance:*

$$\mathbb{E}[\|\mathbf{x}_i^{lm} - \mathbf{x}_j^{lm}\|^2] \leq \mathbb{E}[\|\mathbf{x}_i^{(0)} - \mathbf{x}_j^{(0)}\|^2] - \delta, \quad (\delta > 0) \tag{21}$$

then the Demographic Parity gap (DP) can worsen. This increase is related to the number of layers K, the dominant eigenvalue λ_1 of $\hat{\mathbf{A}}$, and δ:

$$DP^{lm} \geq DP^{(0)} + C \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k} \tag{22}$$

where C is a positive constant.

Proof. We model the GNN using a K-layer GCN. Let the GNN layer operation be defined as:

$$\mathbf{H}^{(k)} = \sigma(\hat{\mathbf{A}}\mathbf{H}^{(k-1)}\mathbf{W}^{(k)}) \tag{23}$$

For simplification, we assume a linear GCN, meaning the activation function $\sigma(x) = x$ and the weight matrices $\mathbf{W}^{(k)} = \mathbf{I}$. Under these assumptions, the output embeddings after K layers are $\mathbf{H}^{(K)} = \hat{\mathbf{A}}^K\mathbf{X}^{\text{lm}}$. Our linear GCN model intentionally isolates this core mechanism. By removing the non-linear activation function σ (e.g., ReLU) for the theoretical analysis, we can obtain a closed-form solution that clearly shows how the graph's spectral properties ($\hat{A}^K$) directly act upon the initial biased features (X_0) (Wu et al., 2019; Cai, 2023). The primary role of the non-linear activation σ is to increase the model's expressive power to learn complex decision boundaries for the downstream task. However, it does not negate the underlying structural aggregation that happens before the activation is applied. In fact, if the task labels are correlated with the homophilous structure (which they often are), the non-linear function will likely learn to reinforce the clustering introduced by the message passing, rather than counteract it. (Wang et al., 2021). Therefore, the linear model can be seen as capturing the fundamental source of the bias amplification, providing a clear and interpretable lower bound on the effect. The link prediction probability between nodes u and v with final embeddings $\mathbf{z}_u, \mathbf{z}_v$ (from $\mathbf{H}^{(K)}$) is given by $P(u \sim v) = \sigma(\mathbf{z}_u^T\mathbf{z}_v)$. The Demographic Parity (DP) is defined as the absolute difference in expected link prediction probabilities between intra-class and inter-class pairs:

$$DP = |\mathbb{E}[P(u \sim v)|s_u = s_v] - \mathbb{E}[P(u \sim v)|s_u \neq s_v]| \tag{24}$$

The covariance of the K-layer output embeddings can be related to the input covariance:

$$\text{Cov}(\mathbf{H}^{(K)}) = \hat{\mathbf{A}}^K\text{Cov}(\mathbf{X}^{\text{lm}})(\hat{\mathbf{A}}^K)^T. \tag{25}$$

If we assume that $\hat{\mathbf{A}}^K \approx \lambda_1^K \mathbf{u}_1 \mathbf{u}_1^T$ (due to the dominant eigenvalue)λ_1 approximation),then it follows that:

$$\text{Cov}(\mathbf{H}^{(K)}) \approx \lambda_1^{2K} \text{Cov}(\mathbf{X}^{\text{lm}}) \tag{26}$$

. The initial assumption about reduced distance for same-attribute pairs implies a relationship for their covariance. Specifically, if we align by the mean of similar nodes, the trace of their covariance satisfies:

$$\text{tr}(\text{Cov}(\mathbf{x}_i^{\text{lm}}, \mathbf{x}_j^{\text{lm}})) \geq \text{tr}(\text{Cov}(\mathbf{x}_i^{(0)}, \mathbf{x}_j^{(0)})) + \delta/2 \quad \text{for } s_i = s_j. \tag{27}$$

Let $\delta_{\text{Cov}}^{\text{lm}}$ denote the difference between the trace of intra-class covariance and inter-class covariance for LM features:

$$\delta_{\text{Cov}}^{\text{lm}} \coloneqq \text{tr}(\text{Cov}(\mathbf{X}^{\text{lm}})_{\text{intra}}) - \text{tr}(\text{Cov}(\mathbf{X}^{\text{lm}})_{\text{inter}}). \tag{28}$$

Similarly, let $\delta_{\text{Cov}}^{(0)}$ be this difference for the initial features $\mathbf{X}^{(0)}$:

$$\delta_{\text{Cov}}^{(0)} \coloneqq \text{tr}(\text{Cov}(\mathbf{X}^{(0)})_{\text{intra}}) - \text{tr}(\text{Cov}(\mathbf{X}^{(0)})_{\text{inter}}). \tag{29}$$

The problem's assumption implies that the LM-induced contraction δ increases this covariance difference:

$$\delta_{\text{Cov}}^{\text{lm}} \geq \delta_{\text{Cov}}^{(0)} + C_1 \delta \quad \text{for some constant } C_1 > 0. \tag{30}$$

The expected inner product difference between intra-class and inter-class pairs, ΔE, is crucial for DP:

$$\Delta E \coloneqq \mathbb{E}[\mathbf{z}_u^T \mathbf{z}_v | s_u = s_v] - \mathbb{E}[\mathbf{z}_u^T \mathbf{z}_v | s_u \neq s_v]. \tag{31}$$

This difference can be decomposed into mean and covariance components:

$$\Delta E = (\boldsymbol{\mu}_{\text{intra}}^{(K)T} \boldsymbol{\mu}_{\text{intra}}^{(K)} - \boldsymbol{\mu}_{s_a}^{(K)T} \boldsymbol{\mu}_{s_b}^{(K)}) + (\text{tr}(\text{Cov}(\mathbf{H}^{(K)})_{\text{intra}}) - \text{tr}(\text{Cov}(\mathbf{H}^{(K)})_{\text{inter}})). \tag{32}$$

Assuming a layer-wise accumulation model for covariance amplification consistent with the theorem's summation $\sum \lambda_1^{2k}$ (which reflects the iterative nature of GNN message passing where bias can accumulate at each layer), we approximate ΔE:

$$\Delta E \approx \Delta \boldsymbol{\mu}_K + \left(\sum_{k=1}^{K} \lambda_1^{2k} \right) (\delta_{\text{Cov}}^{(0)} + C_1 \delta). \tag{33}$$

The DP gap is related to ΔE through the sigmoid function's derivative:

$$\text{DP}^{\text{lm}} \approx |\sigma'(\xi)| \cdot |\Delta E|, \quad \text{where } \xi \text{ is an intermediate point.} \tag{34}$$

Substituting the expression for ΔE and assuming the bias terms accumulate:

$$\text{DP}^{\text{lm}} \geq |\sigma'(\xi)| \cdot |\Delta \boldsymbol{\mu}_K + (\sum_{k=1}^{K} \lambda_1^{2k}) \delta_{\text{Cov}}^{(0)}| + |\sigma'(\xi)| \cdot C_1 \delta \cdot \left(\sum_{k=1}^{K} \lambda_1^{2k} \right). \tag{35}$$

This gives the final lower bound, where $\text{DP}^{(0)}$ represents the base level of disparity and $C = |\sigma'(\xi)| C_1$:

$$\text{DP}^{\text{lm}} \geq \text{DP}^{(0)} + C \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k}. \tag{36}$$

□

B.2 Effectiveness of FSED

Theorem B.2. *FSED corrects the covariance structure of the LM feature distribution to suppress intra-class over-clustering. The lower bound in Theorem B.1 is corrected as:*

$$DP^{lm_{sel}} \leq DP^{lm} - C \cdot \alpha \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k} \tag{37}$$

where $\alpha > 0$ is a variance recovery coefficient.

Proof. FSED modifies the training set by selectively dropping edges. Let L' be the Empirical Risk (ER) on this FSED-modified set. The phenomenon of simplicity bias suggests that models often find it easier to fit "simpler" or more common patterns. If $Q(e)$ represents model error or difficulty for edge e, then for intra-class positive ($+$) and negative ($-$, harder) samples:

$$\frac{\sum_{e\sim\mathcal{E}^+_{\text{intra}}} Q(e)}{|\mathcal{E}^+_{\text{intra}}|} < \frac{\sum_{e\sim\mathcal{E}^-_{\text{intra}}} Q(e)}{|\mathcal{E}^-_{\text{intra}}|}. \tag{38}$$

FSED aims to upweight these harder samples (or, equivalently, downweight easier ones).

The gradient of the training loss $\mathcal{L}_1$ (which includes a Cross-Entropy term for selected edges $\mathcal{E}_{\text{sel}}$ and a KL-Divergence term for unselected edges $\mathcal{E}_{\text{un}}$) with respect to model parameters θ is:

$$\nabla_\theta \mathcal{L}_1 = \sum_{e\in\mathcal{E}_{\text{sel}}} \frac{\partial \text{CE}}{\partial \theta} + \sum_{e\in\mathcal{E}_{\text{un}}} \frac{\partial \mathbb{D}_{\text{KL}}}{\partial \theta}. \tag{39}$$

When considering $\mathcal{E}^-_{\text{intra}}$, the gradient direction aims to increase its distance from the class center, thereby increasing intra-class variance. This effective modification of the intra-class covariance by FSED can be quantified as:

$$\mathbf{\Sigma}_s^{\text{lm}_{\text{sel}}} = \mathbf{\Sigma}_s^{\text{lm}} + \alpha \cdot \delta \cdot \mathbf{I}, \tag{40}$$

where α is the variance recovery coefficient. It reflects the increased proportion of "hard" samples in the selected set:

$$\alpha = \frac{|\mathcal{E}^-_{\text{intra}} \cap \mathcal{E}_{\text{sel}}|}{|\mathcal{E}_{\text{intra}} \cap \mathcal{E}_{\text{sel}}|}. \tag{41}$$

We assume $\alpha > 1$ as FSED prioritizes these harder samples, leading to variance recovery.

The problematic component of the covariance difference, $\delta^{\text{lm}}_{\text{Cov}}$, which contributes to unfairness due to the initial contraction δ, is reduced for the selected set $\mathcal{E}_{\text{sel}}$:

$$\delta^{\text{lm}}_{\text{Cov,sel}} \approx \delta^{\text{lm}}_{\text{Cov}} - C_2\alpha\delta \quad \text{for some constant } C_2 > 0. \tag{42}$$

This means FSED counteracts a portion of the bias. The expected inner product difference for the selected set, ΔE_{sel}, is then:

$$\Delta E_{\text{sel}} \approx \Delta\boldsymbol{\mu}_K + \left(\sum_{k=1}^{K} \lambda_1^{2k}\right)(\delta^{\text{lm}}_{\text{Cov}} - C_2\alpha\delta). \tag{43}$$

This can be rewritten in terms of the original ΔE:

$$\Delta E_{\text{sel}} \approx \Delta E - \left(\sum_{k=1}^{K} \lambda_1^{2k}\right) C_2\alpha\delta. \tag{44}$$

The DP for the selected set is $\text{DP}^{\text{lm}_{\text{sel}}} \approx |\sigma'(\xi')| \cdot |\Delta E_{\text{sel}}|$. Assuming the reduction effectively decreases the disparity:

$$\text{DP}^{\text{lm}_{\text{sel}}} \leq |\sigma'(\xi')| \cdot |\Delta E| - |\sigma'(\xi')| \cdot C_2\alpha\delta \cdot \left(\sum_{k=1}^{K} \lambda_1^{2k}\right). \tag{45}$$

This leads to the corrected upper bound for DP after FSED, where $C = |\sigma'(\xi')|C_2$:

$$\mathrm{DP}^{\mathrm{lm}_{\mathrm{sel}}} \leq \mathrm{DP}^{\mathrm{lm}} - C \cdot \alpha \cdot \delta \cdot \sum_{k=1}^{K} \lambda_1^{2k}. \tag{46}$$

□

B.3 Derivation of ANPO

Theorem B.3. *The ANPO loss (e.g., Eq. 15 for $\mathcal{L}_{pos}$) effectively models edge generation patterns through preference optimization.*

Proof. Consider a preferred edge $e_w = (v, w)$ and a dispreferred edge $e_l = (v, l)$ relative to an anchor node v and label y. The probability that the model prefers e_w over e_l according to an optimal reward function r^* is given by the Bradley-Terry model:

$$p^*(P(e_w) > P(e_l)) = \frac{\exp(r^*(e_w, y))}{\exp(r^*(e_w, y)) + \exp(r^*(e_l, y))}. \tag{47}$$

The Anchor Node Preference Optimization (ANPO) aims to train a policy P_θ to align with these preferences, typically by maximizing the expected log-likelihood of observed preferences, often regularized by a KL divergence from a reference policy P_{ref}:

$$\mathbb{E}_{y\sim\text{True},e_w,e_l\sim\mathcal{E}_v^+}[\log \sigma(r_\theta(e_w, y) - r_\theta(e_l, y))] \tag{48}$$

Following the Direct Preference Optimization (DPO) framework, we seek to optimize a policy $\psi_\theta(e|y)$ (representing $P_\theta(e)$) based on preferences induced by a latent reward $r(e, y)$. The objective often involves maximizing $\mathbb{E}_{y\sim\text{True},e\sim\mathcal{E}_v^+}[r_\phi(e, y) - \beta \log \frac{\psi_\theta(e|y)}{P_{\text{ref}}(e|y)}]$.

The optimal policy $\psi^*(e|y)$ under such a framework can be expressed in terms of the reference policy $\psi_{\text{ref}}(e|y)$ and the true reward $r^*(e, y)$:

$$\psi^*(e|y) = \frac{1}{Z(y)}\psi_{\text{ref}}(e|y) \exp\left(\frac{1}{\beta}r^*(e, y)\right), \tag{49}$$

where $Z(y)$ is a normalization constant (partition function):

$$Z(y) = \sum_{e'} \psi_{\text{ref}}(e'|y) \exp\left(\frac{1}{\beta}r^*(e', y)\right). \tag{50}$$

From this, the implicit optimal reward $r^*(e, y)$ can be related to the optimal and reference policies:

$$r^*(e, y) = \beta\left(\log \frac{\psi^*(e|y)}{\psi_{\text{ref}}(e|y)} + \log Z(y)\right). \tag{51}$$

Substituting this expression for $r^*(e, y)$ back into the Bradley-Terry model for preferences $p^*(P(e_w) > P(e_l))$, we arrive at the characteristic DPO loss form, which ANPO utilizes:

$$p^*(P(e_w) > P(e_l)) = \frac{1}{1 + \exp\left(\beta\left(\log \frac{\psi^*(e_l|y)}{\psi_{\text{ref}}(e_l|y)} - \log \frac{\psi^*(e_w|y)}{\psi_{\text{ref}}(e_w|y)}\right)\right)}. \tag{52}$$

Minimizing the negative log-likelihood of this preference probability (i.e., $-\log p^*$) yields the ANPO loss for a pair of preferred/dispreferred edges. The loss for $\mathcal{L}_{\text{neg}}$ follows a similar derivation. □

B.4 Effectiveness of ANPO

Theorem B.4. *After ANPO, node representations can compress inter-class mean differences, thereby alleviating unfairness.*

Proof. We consider the effect of the $\mathcal{L}_{\text{pos}}$ component of the ANPO loss on the embedding $\mathbf{z}_v$ of an anchor node v. Let $e_w = (v, w)$ be a preferred edge (e.g., an inter-class connection that should be strengthened) and $e_l = (v, l)$ be a dispreferred one (e.g., an intra-class connection for a boundary node). The gradient of $\mathcal{L}_{\text{pos}}$ with respect to $\mathbf{z}_v$ is:

$$\frac{\partial \mathcal{L}_{\text{pos}}}{\partial \mathbf{z}_v} \propto -\beta \cdot \sigma(-\Delta) \cdot (\mathbf{z}_w - \mathbf{z}_l), \tag{53}$$

where Δ is the difference in scaled log-probability ratios between e_w and e_l:

$$\Delta = \beta \left(\log \frac{\psi_\theta(e_w|y)}{\psi_{\text{ref}}(e_w|y)} - \log \frac{\psi_\theta(e_l|y)}{\psi_{\text{ref}}(e_l|y)} \right). \tag{54}$$

If e_w is indeed preferred by the current policy ψ_θ relative to ψ_{ref} more than e_l is, then $\Delta > 0$. The term $\sigma(-\Delta)$ will be less than 0.5. The negative sign in the gradient indicates that $\mathbf{z}_v$ will be updated to decrease the loss, effectively moving it "towards" $\mathbf{z}_w$ and "away from" $\mathbf{z}_l$.

More directly, the DPO framework aims to increase the likelihood of e_w and decrease that of e_l. If $P(e_x) = \sigma(\mathbf{z}_v^T \mathbf{z}_x)$, the update for $\mathbf{z}_v$ will be in a direction that makes $\mathbf{z}_v$ more aligned with $\mathbf{z}_w$ and less aligned with $\mathbf{z}_l$:

$$\mathbf{z}_v^{\text{new}} = \mathbf{z}_v^{\text{old}} + \eta_{\text{ANPO}}(\mathbf{z}_w - \mathbf{z}_l), \tag{55}$$

where η_{ANPO} is an effective learning rate.

Consider a node v belonging to sensitive group S_0. If ANPO is designed to encourage inter-class connections for fairness, then for such a v, the preferred neighbor w might often belong to group S_1, while a dispreferred (positive but less desired) neighbor l might belong to S_0. On average, $\mathbf{z}_w$ would be close to the mean $\boldsymbol{\mu}_1$ of group S_1, and $\mathbf{z}_l$ close to $\boldsymbol{\mu}_0$. The update for $\mathbf{z}_v$ can then be approximated as a shift relative to these means:

$$\mathbf{z}_v^{\text{new}} \approx \mathbf{z}_v^{\text{old}} + \gamma(\boldsymbol{\mu}_1^{\text{old}} - \boldsymbol{\mu}_0^{\text{old}}), \tag{56}$$

where γ is an effective coefficient related to η_{ANPO} and the prevalence/strength of such inter-class preferences (e.g., $\gamma = \beta \frac{|\mathcal{E}^+_{v,\text{inter}}|}{|\mathcal{E}_v|}$ as suggested in the paper).

This local update rule affects the global mean embeddings of the sensitive groups. Let $\Delta\boldsymbol{\mu} := \boldsymbol{\mu}_0^{\text{new}} - \boldsymbol{\mu}_1^{\text{new}}$ be the new difference between group means. The new mean for group S_0 is approximately:

$$\boldsymbol{\mu}_0^{\text{new}} \approx \boldsymbol{\mu}_0^{\text{old}} + \bar{\gamma}_0(\boldsymbol{\mu}_1^{\text{old}} - \boldsymbol{\mu}_0^{\text{old}}), \quad \text{where } \bar{\gamma}_0 \text{ is an average } \gamma. \tag{57}$$

Similarly, for group S_1, if ANPO symmetrically encourages connections to S_0:

$$\boldsymbol{\mu}_1^{\text{new}} \approx \boldsymbol{\mu}_1^{\text{old}} + \bar{\gamma}_1(\boldsymbol{\mu}_0^{\text{old}} - \boldsymbol{\mu}_1^{\text{old}}). \tag{58}$$

The difference between these new means is:

$$\Delta\boldsymbol{\mu} \approx (\boldsymbol{\mu}_0^{\text{old}} - \boldsymbol{\mu}_1^{\text{old}})(1 - \bar{\gamma}_0 - \bar{\gamma}_1). \tag{59}$$

If $0 < \bar{\gamma}_0 + \bar{\gamma}_1 < 1$, then $1 - \bar{\gamma}_0 - \bar{\gamma}_1$ is positive and less than 1. This implies that the magnitude of the inter-class mean difference $\|\Delta\boldsymbol{\mu}\|$ is reduced, signifying a compression of these differences and thus an alleviation of this aspect of unfairness. □

C Supplementary Materials and Results

C.1 Benchmarks

The statistics of four deployed benchmark datasets are lited below.

Table 5: Statistics of used four benchmark datasets.

Dataset	#Nodes	#Edges	#Category	# Feature	# Sensitive Feature
Cora	2,708	5,429	7	1433	Paper category
CiteSeer	3,327	9,104	6	3703	Paper category
PubMed	19,717	88,648	3	500	Paper category
Pokec-n	66,569	729,129	2	265	Region
Pokec-z	67,797	882,765	4	277	Region

Table 6: Results comparison on Pokec-z.

EM	GCN							GAT						
	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
Link Prediction on Pokec-z.														
SE	80.71±0.33	60.13±0.99	17.11±1.34	10.98±2.12	7.56±1.11	10.71±2.04	22.33±1.96	79.72±0.61	58.24±1.12	18.33±1.45	11.45±1.87	7.92±0.98	10.33±1.76	21.87±1.82
Vanilla	81.88±0.57	61.11±1.51	18.99±0.97	17.71±1.51	12.81±3.17	12.11±2.51	23.18±2.06	80.15±0.37	61.87±1.42	19.45±1.12	16.82±1.33	11.92±2.89	11.76±2.32	22.64±1.97
GLEM	82.47±0.81	59.42±0.95	19.76±1.15	15.43±0.59	8.81±1.43	11.48±1.99	24.54±2.56	82.01±0.92	57.63±1.08	20.11±1.27	14.92±0.72	8.45±1.28	8.02±1.76	20.87±2.41
SimTEG	82.21±1.01	61.56±1.41	17.98±1.59	12.77±1.42	8.98±2.02	10.65±0.79	22.11±2.31	81.68±0.41	60.12±1.35	18.45±1.42	13.24±1.27	8.67±1.87	10.12±0.86	21.56±2.18
FairTAG	83.89±0.78	53.31±0.92	14.11±0.71	9.15±0.65	8.78±1.42	8.96±1.82	18.33±1.93	82.87±0.66	52.45±0.87	13.76±0.82	8.92±0.72	5.32±1.35	8.15±1.67	17.89±1.76

Table 7: Performance comparison with GNN side fairness-aware methods on Pokec-z benchmark.

Method	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)
Link Prediction on Pokec-z					
FairDrop	81.0±0.7	58.0±1.2	25.0±2.0	15.0±1.5	10.5±1.3
DropEdge	81.3±0.6	58.5±1.1	24.5±2.2	15.5±1.7	11.0±1.4
FairGT	82.5±0.8	55.0±1.3	18.0±1.8	11.5±1.3	7.5±1.0
G-FAME++	83.3±0.9	55.5±1.4	17.5±1.7	11.0±1.1	7.0±1.0
FairTAG	83.9±0.8	53.3±0.9	14.1±0.7	9.2±0.7	8.8±1.4

C.2 Results on Pokec-z

We supplement the results from Pokec-z in this section, including comparison with LM-as-encoder and GNN side fairness-aware methods. As demonstrated in Table 6 and Table 7, the advantage of FairTAGextends to Pokec-z benchmark.

C.3 Ablation Study on Training Stages

In this section, we conduct ablation studies on the proposed two-stage optimization strategy. The two variants of FairTAG are: 1) For $\mathcal{F}_{ref}$ after using the FSED, we do not employ ANPO to do subsequent optimization; 2) We do not use FSED, but directly perform ANPO based on $\mathcal{F}_{oracle}$.

The results in Table 8 indicate that the two-stage alignment is highly effective. Firstly, except for the complete FairTAG on the PubMed dataset showing a slight weakness in the AUC metric, the performance on the other three datasets has been optimized. Additionally, the two-stage alignment has sequentially optimized the fairness metrics in all but a few cases. Taking the ΔDP_m and ΔEO_m metrics as an example, on the Cora dataset, the fairness score-based edge drop strategy and the ANPO singlely achieve optimizations of 6.7% and 3.2% on ΔDP_m, and 11.2% and 9.7% on ΔEO_m, respectively. And On the PubMed dataset, the two stage improvement is 3.9% and 18.2% on the ΔDP_m, and 28.6% and 17.1% on the ΔEO_m.

C.4 Generalizability on GNN Scale

We fix the best checkpoint of the LM encoder and further conduct a broader parameter search for the GNN component. Learning rate, batch size and training epoch are fixed at 1e-4, 256 and 500. The experiments on Pokec-n are shown below:

C.5 Generalizability on LM Backbone

We additionally provide experiments on sentence-t5-large (Table 10) to validate the generalizability of FairTAG.

Table 8: Ablation study results on four selected benchmarks (*Cora*, *CiteSeer*, *PubMed* and *Pokec-n*). We sequentially discard modules from the back to the front to verify the effectiveness of the two-stage optimization. **Highlight** indicates the best performance.

Methods	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
Link Prediction on Cora.							
FairTAG	**93.9±0.6**	**43.6±2.5**	**13.1±4.5**	**11.4±3.9**	**11.3±4.7**	**73.8±3.2**	**100.0±0.0**
FairTAG w/o ANPO	93.3±0.4	47.1±1.9	17.8±3.4	11.8±1.9	15.0±3.1	83.0±3.6	**100.0±0.0**
FairTAG w/o FSED	92.6±0.5	48.9±1.1	18.1±4.6	13.1±3.4	16.2±3.1	83.8±3.0	**100.0±0.0**
Link Prediction on CiteSeer.							
FairTAG	**96.3±0.6**	**42.7±1.5**	**8.7±1.7**	**13.1±3.2**	**9.7±2.0**	**68.6±4.8**	**51.4±18.4**
FairTAG w/o ANPO	95.8±0.5	49.4±1.2	9.6±2.0	13.6±1.8	10.2±1.7	72.6±5.2	57.9±16.8
FairTAG w/o FSED	95.2±0.6	51.5±1.3	11.1±2.4	14.6±2.7	10.7±2.3	73.3±4.9	60.7±15.8
Link Prediction on PubMed.							
FairTAG	**96.0±0.4**	**29.1±1.1**	**9.4±0.8**	**2.2±0.6**	**3.2±0.4**	**21.1±2.0**	**13.9±2.0**
FairTAG w/o ANPO	91.2±0.2	30.3±1.5	11.3±0.8	2.7±0.7	3.4±1.0	21.8±2.2	14.2±1.7
FairTAG w/o FSED	90.9±0.2	35.6±1.7	13.1±1.2	4.1±0.9	3.8±1.1	23.3±1.0	14.4±3.6
Link Prediction on Pokec-n.							
FairTAG	**84.9±0.9**	**47.5±1.3**	**18.7±1.6**	**8.5±1.0**	**4.6±1.1**	**8.5±1.1**	**28.1±2.3**
FairTAG w/o ANPO	83.7±0.2	48.2±1.8	20.0±0.4	9.9±0.8	6.7±1.0	10.9±2.1	34.2±3.7
FairTAG w/o FSED	84.8±1.2	48.9±1.5	21.4±1.4	10.0±0.7	6.1±1.0	12.2±1.5	32.2±2.4

Table 9: Performance comparison across different model hyperparameters. We report AUC (higher is better) and various fairness metrics (lower is better for Δ metrics). Best results in each block are highlighted in **bold**.

Hidden × Layers	**Method**	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
200 × 2	Vanilla	82.5±0.6	60.9±1.5	30.1±2.1	16.9±1.7	11.5±1.5	16.9±1.7	41.9±4.1
	GLEM	83.1±0.7	59.5±1.3	29.8±2.5	15.8±1.8	10.9±1.3	15.8±1.8	40.8±3.8
	SimTEG	83.9±0.8	62.1±1.7	29.9±2.1	17.9±1.2	11.8±1.2	16.6±1.7	39.1±3.1
	FairTAG	**84.9±0.9**	**47.5±1.3**	**18.7±1.6**	**8.5±1.0**	**4.6±1.1**	**8.5±1.1**	**28.1±2.3**
800 × 2	Vanilla	82.8±0.5	64.5±1.8	29.0±2.5	15.5±1.2	12.4±2.1	15.8±2.0	40.2±3.5
	GLEM	84.1±0.9	58.5±1.6	30.5±2.7	15.2±1.4	12.2±0.5	16.9±1.9	40.1±2.1
	SimTEG	84.4±0.7	61.0±2.1	30.5±1.8	18.2±1.3	11.7±1.3	16.9±1.6	39.0±3.0
	FairTAG	**85.2±0.8**	**46.3±1.5**	**17.1±1.2**	**9.4±1.4**	**4.6±2.0**	**8.1±1.2**	**29.9±1.4**
200 × 4	Vanilla	82.6±0.7	62.8±1.9	31.5±2.3	17.8±2.0	12.3±1.8	17.9±1.9	43.2±4.3
	GLEM	83.4±0.9	61.2±1.7	31.2±2.8	16.9±2.1	11.8±1.5	18.7±2.0	42.1±4.1
	SimTEG	84.2±0.9	63.9±2.1	31.4±2.4	19.0±1.6	12.7±1.5	17.5±1.9	41.6±3.5
	FairTAG	**85.3±0.4**	**48.9±1.9**	**19.8±1.9**	**9.4±1.3**	**5.5±1.4**	**9.5±1.4**	**29.7±2.6**
200 × 8	Vanilla	82.9±0.8	64.0±2.2	33.0±2.7	18.8±2.1	13.2±1.9	18.9±2.2	44.6±4.7
	GLEM	83.6±0.9	62.7±2.0	32.5±3.1	17.9±2.3	12.6±1.8	17.8±2.3	43.5±4.5
	SimTEG	84.3±1.1	65.4±2.3	32.7±2.7	19.9±1.9	13.3±1.7	18.4±2.1	45.6±3.8
	FairTAG	**85.8±1.1**	**50.0±1.9**	**20.8±2.2**	**10.2±1.6**	**6.3±1.7**	**10.4±1.6**	**33.1±3.1**

C.6 Ablation on Integration of GCN

In fact, since our LM has been trained on graph-related tasks and has already demonstrated good usability, we have removed the integration of GCN to further observe more conclusions. As demonstrated in Table 11, adding a GCN decoder can still provide better link prediction performance, but the downside is that fairness often decreases in most cases.

D Sampling Procedure

In this section, we give a supplementary explain fo training data sampling introduced in Section 1 and Section 5. As in previous work (Liu et al., 2023; Spinelli et al., 2021), we sample negative samples for training with the same number of positive samples, using the sampling code from

Table 10: Main results for link prediction on four benchmark datasets. We compare our FairTAG with several baselines. Best results for each metric within a dataset are highlighted in **bold**.

Method	AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
Link Prediction on Cora							
SE	91.3	50.1	17.1	18.0	21.8	88.2	**100.0**
Vanilla	89.1	53.0	25.1	17.2	20.6	86.5	**100.0**
GLEM	93.9	54.5	23.9	18.9	24.3	91.9	**100.0**
SimTEG	93.4	51.9	20.5	14.2	18.0	89.2	**100.0**
FairTAG	**94.7**	**41.8**	**11.8**	**10.5**	**10.1**	**71.9**	**100.0**
Link Prediction on CiteSeer							
SE	94.5	55.8	38.1	20.4	12.8	80.1	71.9
Vanilla	95.3	56.9	22.1	22.8	11.5	82.9	65.7
GLEM	**96.3**	58.1	22.9	20.1	14.2	84.1	69.5
SimTEG	96.0	55.8	12.1	15.5	12.4	74.8	61.1
FairTAG	**96.3**	**40.9**	**7.5**	**12.1**	**8.5**	**66.9**	**49.8**
Link Prediction on PubMed							
SE	95.1	45.9	16.5	5.0	4.2	59.2	34.0
Vanilla	95.3	39.4	12.9	6.3	5.3	37.5	20.9
GLEM	96.2	40.8	14.9	6.6	5.6	39.8	20.1
SimTEG	96.2	38.1	15.6	7.3	5.9	29.1	16.9
FairTAG	**97.0**	**27.5**	**8.2**	**1.7**	**2.6**	**19.5**	**12.1**
Link Prediction on Pokec-n							
SE	82.5	56.8	27.2	13.5	8.2	13.5	37.0
Vanilla	83.5	59.9	29.5	16.4	11.2	16.5	41.1
GLEM	84.2	58.7	29.0	15.2	10.5	15.1	40.2
SimTEG	**86.9**	61.2	29.3	17.5	11.6	16.1	38.5
FairTAG	85.9	**45.9**	**17.5**	**7.5**	**3.9**	**7.6**	**26.8**

Table 11: Ablation Study: Impact of GCN Component

		AUC (↑)	ΔDP_m (↓)	ΔEO_m (↓)	ΔDP_g (↓)	ΔEO_g (↓)	ΔDP_s (↓)	ΔEO_s (↓)
Cora	w/ GCN	**93.9±0.6**	43.6±2.5	13.1±4.5	**11.4±3.9**	11.3±4.7	73.8±3.2	100.0±0.0
	w/o GCN	92.9±0.5	**40.5±2.2**	**12.1±4.2**	13.4±3.5	**10.8±4.5**	**71.5±3.0**	100.0±0.0
CiteSeer	w/ GCN	**96.3±0.6**	42.7±1.5	8.7±1.7	13.1±3.2	9.7±2.0	68.6±4.8	51.4±18.4
	w/o GCN	95.3±0.7	**40.1±1.2**	**8.1±1.5**	**12.0±3.0**	**9.1±1.8**	**66.5±4.2**	**48.5±16.5**
PubMed	w/ GCN	**96.0±0.4**	29.1±1.1	9.4±0.8	2.2±0.6	3.2±0.4	**21.1±2.0**	13.9±2.0
	w/o GCN	90.5±0.7	**28.0±1.0**	**8.8±0.7**	**2.0±0.5**	**2.9±0.3**	22.3±1.8	**13.0±1.8**
Pokec-n	w/ GCN	**84.9±0.9**	47.5±1.3	18.7±1.6	8.5±1.0	**4.6±1.1**	8.5±1.1	28.1±2.3
	w/o GCN	82.9±1.0	**43.5±1.0**	**18.1±1.5**	**8.0±0.9**	5.1±0.9	**8.1±1.0**	**27.0±2.0**

torch_geometric.utils.negative_sampling. The specifics are as follow:

```
neg_edges_train = negative_sampling(
edge_index=train_pos_edge_index,
num_nodes=num_nodes,
num_neg_samples=train_pos_edge_index.size(1)
)
```

E Evaluation Metrics

In this section, we carefully introduce the metrics used. The AUC is a metric used to evaluate the performance of a classification model across different classification thresholds. It is particularly useful in binary classification problems and is represented by the area under the Receiver Operating Characteristic (ROC) curve. The AUC is mathematically defined as the integral of the True Positive Rate (TPR) with respect to the False Positive Rate (FPR), as shown in the following formula:

$$\text{AUC} = \int_0^1 \text{TPR}\, d(\text{FPR}) \tag{60}$$

The True Positive Rate (TPR), also known as sensitivity, is the ratio of true positive predictions to the total actual positives and is given by:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{61}$$

The False Positive Rate (FPR), which is the ratio of false positive predictions to the total actual negatives, is defined as:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{62}$$

In these formulas: - TP is the number of true positives, where the model correctly predicted the positive class. - FN is the number of false negatives, where the model incorrectly predicted the negative class for a positive instance. - FP is the number of false positives, where the model incorrectly predicted the positive class for a negative instance. - TN is the number of true negatives, where the model correctly predicted the negative class.

The AUC value ranges between 0 and 1, where an AUC of 1 indicates a perfect classifier, and an AUC of 0.5 suggests a model with no discriminative power, performing no better than random guessing.

Then, we introduce the fairness metrics used.

Demographic Parity (DP): Demographic parity is a fairness criterion in machine learning that requires the distribution of positive outcomes to be the same across different demographic groups. Mathematically, this means that the probability of a positive outcome is independent of the group membership. Lets define Y as the binary label indicating the favorable outcome, A as the sensitive attribute indicating group membership. $\hat{Y}$ as the predicted outcome from a model. DP requires:

$$P(\hat{Y}|A = 0) = P(\hat{Y}|A = 1) \tag{63}$$

Equalized Odds (EO): EO is a term often used in the context of fairness in machine learning and statistical analysis. It refers to a condition where the true positive rate and the true negative rate are equal across different groups or classes.

Continuing from the above definition, then EO requires:

$$P(\hat{Y} = 1|A = 0, Y = y) = P(\hat{Y} = 1|A = 1, Y = y) \tag{64}$$

For the task of link prediction, there are many ways to categorize edges. The author of (Spinelli et al., 2021) proposed a set of widely accepted classification criteria:

- **Mixed dyadic** (Spinelli et al., 2021): This type decides two dyadic groups, in which one category of edges connects nodes that have the same fairness attribute, and these are regarded as intra-relations. The other category of edges connects nodes that have different fairness attributes, and these are regarded as inter-relations. This distinction is crucial in the analysis of network structures, as it helps to understand the dynamics within homogeneous and heterogeneous groups in terms of fairness.
- **Sub-group dyadic** (Spinelli et al., 2021): This criterion treats all connections within groups that possess fairness attributes as a single set, which means that inter-relations are divided into finer-grained subgroups. It aims to ensure a balance between intra-relations and all inter-relations.

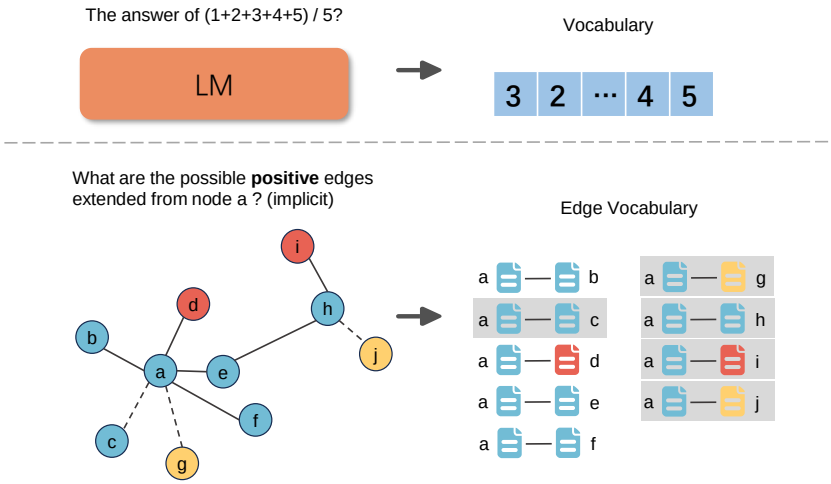

Figure 6: An Analogous Introduction to ANPO.

- **Group dyadic** (Spinelli et al., 2021): This dimension indicates a one-to-one mapping from dyadic to node-level fairness attributes. An edge is categorized into a specific group based on the sensitive attributes it contains. This binary definition ensures that all nodes, regardless of the magnitude of their sensitive attributes, participate in the establishment of edges.

F Hyperparameters Search Space

In this section, we demonstrate the search for important parameters on LMs and GNNs in Table 12.

Table 12: Hyperparameters search spaces used in experiments.

LM		GNN	
Hyperparameter	Search Space	Hyperparameter	Search Space
learning rate	[1e-3, 1e-4]	hidden size	200
epoch	[5, 10]	layers	2
max length	512	GAT heads	4
batch size	16	dropout	0.2
β	[0.1, 0.3]	learning rate	[1e-3, 1e-4]
λ	[10, 30]	epoch	[500, 1000]
γ	[0.2, 0.3, 0.4]	batch size	256

G Supplementary Explanation for ANPO

As demonstrated in Figure 6, we use the generation process of LMs to analogize the edge generation process based on the anchor node. In the context of LMs, when we are given a question like 'The answer of (1+2+3+4+5) / 5?', the model will generate the probability of the next token on the vocabulary. For the local structure of the anchor node, assuming our current task is to determine the existence of positive edges, we are actually implicitly asking 'What are the possible positive edges extended from node a?' Of course, we do not construct prompts with local information as in previous work to accomplish this. Our vocabulary theoretically includes all the edges on the graph. However, we do not add negative edges and non-connected edges as noise to the preference optimization, that is, we directly set the scores of these 'edge vocabulary' to 0 (the grayed-out parts in the Figure 6). Furthermore, we have optimized Equation 15 and Equation 16, allowing our e_w to select only the highest scorer from $\mathcal{E}^{+}_{v,inter}$ and the lowest scorer from $\mathcal{E}^{-}_{v,intra}$.

H Notations

In this section, we present a summary description of the notation from Section 4 in Table 13.

Table 13: Notations used in proposed methodology.

Notation	Description
e_i	A single edge i in training set.
t_{i_src},t_{i_dst}	Textual descriptions of source node and destination node of edge i.
$\mathcal{F}_{oracle}$	Oracle model that finetunes on total training set.
$\mathcal{E}$	Total training edge set.
$\mathcal{E}_{intra}, \mathcal{E}_{inter}$	Training edge set only including intra-relation and inter-relation respectively, as described in Section 3.
$\mathcal{E}^+_{intra}$, $\mathcal{E}^-_{intra}$	Positive and negative samples in $\mathcal{E}_{intra}$.
$\mathcal{E}^+_{inter}$, $\mathcal{E}^-_{inter}$	Positive and negative samples in $\mathcal{E}_{inter}$.
$\mathcal{V}^+$	Anchor nodes that has both connected relations in $\mathcal{E}^+_{intra}$ and $\mathcal{E}^+_{inter}$.
$\mathcal{V}^-$	Anchor nodes that has both connected relations in $\mathcal{E}^-_{intra}$ and $\mathcal{E}^-_{inter}$.
$\mathcal{E}^+_{v,intra}$,$\mathcal{E}^+_{v,inter}$	From node v, the positive examples in intra-relations and inter-relations that are extended.
$\mathcal{E}^-_{v,intra}$,$\mathcal{E}^-_{v,inter}$	From node v, the negative examples in intra-relations and inter-relations that are extended.