Mitigating Bias for Unseen Demographic Groups in Graph Neural Networks

Francisco Santos, Pang-Ning Tan and Abdol-Hossein Esfahanian

Department of Computer Science and Engineering, Michigan State University, Lansing, 48824, MI, United States of America.

*Corresponding author(s). E-mail(s): santosf3@msu.edu;
Contributing authors: ptan@msu.edu; esfahanian@msu.edu;

Abstract

Fairness is an important factor to consider in graph neural networks (GNNs) as biases in the data can be amplified by the link structure. Despite ongoing research, existing fairness-aware GNN methods often assume that the sensitive attribute values for all demographic groups are available during training. This assumption restricts their practical applicability, especially in scenarios where training examples for certain demographic groups are unavailable. To address this limitation, we propose FairGRUNT, a novel GNN framework designed to handle training scenarios with unseen demographic groups. FairGRUNT employs disentangled representation learning to separate node embeddings for class prediction from those encoding demographic information, thereby reducing dependency between them. A pretraining stage assigns soft labels to a subset of the unlabeled nodes indicating seen or unseen group membership based on their prediction confidence. These soft labels are then used to train a demographic classifier and guide a statistical parity-based fairness regularizer, which is integrated into the training objective to mitigate bias in the GNN predictions. Experimental results on real-world datasets show that FairGRUNT outperforms traditional fairness-aware GNN methods in reducing biases in node classification, particularly for sensitive attributes with unseen groups.

Keywords: Graph Neural Networks, Algorithmic Bias, Disentangled Representation Learning

1 Introduction

Fairness is essential for graph neural networks (GNNs), as biases present in the data can be amplified through the network structure [1, 2]. Ensuring fairness in GNNs is critical to prevent discrimination based on sensitive attributes such as race, gender, age, or ethnicity. Various approaches have therefore been proposed to improve fairness in graph-based models [2–6]. However, most existing fairness-aware GNN approaches assume full access to protected attribute values during training. This assumption is often unrealistic in real-world scenarios where demographic data may be incomplete or entirely unavailable for certain subpopulations [4]. Since current models rely on the availability of labeled examples from all demographic groups during training, this hinders their ability to generalize debiasing strategies to subgroups unaccounted for in the training data. As a result, fairness performance can degrade significantly when such models encounter previously unseen demographic groups during deployment. Moreover, maintaining a balance between fairness and predictive performance remains difficult, especially when the distributions of seen and unseen groups differ. Our goal is to develop models that promote fairness across all demographic groups, including those not observed during training, while preserving strong predictive performance.

Eliminating biases associated with unseen demographic groups in a GNN is a non-trivial task. First, the GNN model must be capable of separating the feature embeddings associated with the sensitive attribute information from those associated with the prediction task [3, 4]. Our proposed model tackles this challenge by employing a feature disentanglement strategy, explicitly separating the embedding representing the sensitive attribute from the embedding used for the class prediction. By incorporating a feature independence constraint via correlation loss, this ensures that the sensitive attributes do not influence class outcomes, thereby promoting fairness.

The second challenge lies in effectively addressing unseen protected attributes, as merely disentangling the learned representations is insufficient to ensure fair generalization. To address this limitation, we adopt a two-stage strategy. We begin by pretraining a GCN on the portion of the training dataset that contains only seen demographic attributes, with the goal of predicting the corresponding seen demographic groups. After training this model, a subset of the test data—containing both seen and unseen group samples—is passed through. Samples with low prediction confidence are treated as likely belonging to unseen groups and are assigned soft labels to reflect this uncertainty. These soft labels are then used to train a secondary classifier that distinguishes between seen and unseen demographic groups. Once trained, this classifier is applied to the entire dataset to assign a seen or unseen group label to each node. These predicted group labels are subsequently passed to a sensitive attribute classifier, whose outputs are incorporated into a fairness regularizer. This procedure enables the framework to explicitly model demographic group uncertainty and enhances its ability to mitigate bias in predictions across both seen and unseen groups.

The proposed strategies are integrated into a novel GNN framework called **FairGRUNT** (**Fair** **G**raph **R**epresentation learning for **UN**seen sensi**T**ive attribute groups). Experimental results on 3 real-world datasets demonstrate the effectiveness of FairGRUNT in balancing the tradeoff between fairness and performance compared to other fairness-aware GNN methods.

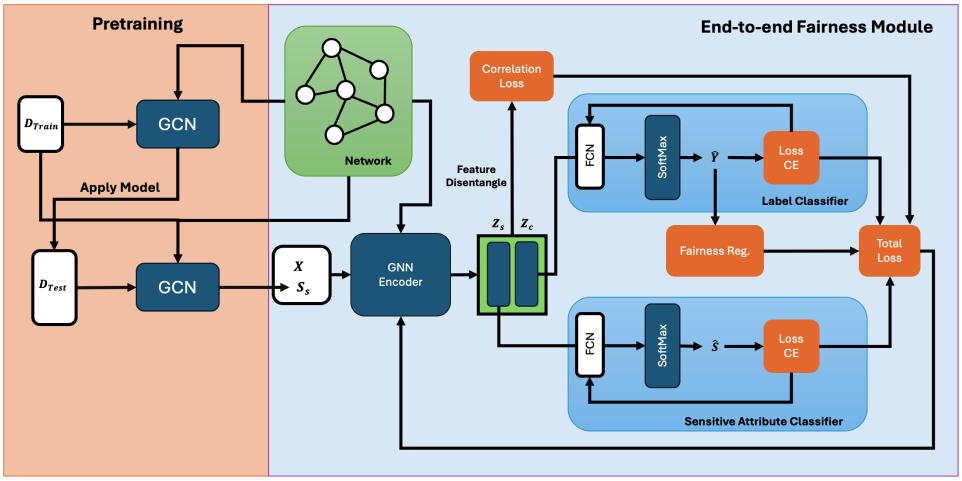

Fig. 1 Framework for FairGRUNT. D_{train} is the training data and D_{test} is the testing data. X represents the feature matrix, while S_s denotes the seen/unseen soft label. $\hat{S}$ indicates the prediction for the seen/unseen attribute, and $\hat{Y}$ represents the class prediction.

2 Preliminaries

Consider a network $G = (V, E, X, Y)$, where V is the set of nodes, $E \subseteq V \times V$ is the set of edges, $X \in \mathbb{R}^{|V| \times d}$ is the node feature matrix, and Y contains the class labels. The feature matrix X consists of two parts: X^p, denoting protected attributes, and X^u, denoting unprotected or task-relevant attributes. We assume that the protected attribute matrix X^p is partially observed, meaning that some nodes have missing or unannotated sensitive attribute values, denoted as $X^p_{\text{unlabeled}}$. Among these, some nodes belong to *seen* demographic groups—those observed during training—while others belong to *unseen* groups, not present in the labeled training data. We define S_s as a soft label indicating whether a node belongs to a seen or unseen group.

Consider the normalized adjacency matrix of the form $\hat{A} = \tilde{D}^{-1/2}(A + I)\tilde{D}^{-1/2}$, where A is the adjacency matrix corresponding to E, I is the identity matrix, and $\tilde{D}$ is the degree matrix of $A + I$. The objective of this work is to design a fairness-aware graph neural network that maintains strong predictive performance while ensuring fair outcomes across all demographic groups, including those unseen during training.

3 Methodology

Figure 1 presents the architecture of FairGRUNT, which comprises of two main components: a pretraining phase and an end-to-end fairness module. In the pretraining phase, we employ a two-step strategy to determine whether each node belongs to a seen or unseen demographic group. A GCN is first trained on nodes with known protected attributes and then applied to test data containing both seen and unseen samples. Nodes with low prediction confidence are treated as unseen and assigned soft labels. A second GCN is trained on these labels to classify all nodes as seen or unseen. The end-to-end fairness module incorporates a GNN encoder that learns two disentangled

embeddings: one for predicting the target label and another for predicting the sensitive attribute. A correlation loss enforces independence between the embeddings to mitigate bias along with a statistical parity-based regularizer. Details are given below.

3.1 Soft Label Generation of Demographic Groups

We introduce a two-stage pretraining strategy that generates soft group labels, enabling the model to differentiate between samples from seen and unseen demographic groups. These soft labels play a crucial role in enforcing fairness constraints in later stages of training. We begin by partitioning the dataset into training and test subsets. We assume that the training set contains samples exclusively from seen demographic groups—those for which labels are available—while the test set contains a mixture of both seen and unseen groups. In the first stage, we train a GCN on the training set to predict the protected attribute among the seen groups. Once trained, this GCN is applied to a subset of the test data in a semi-supervised learning fashion. We compute the model's prediction confidence on each sample. If the model exhibits high confidence, the sample is assumed to belong to a seen group; conversely, if the prediction confidence is low, the sample is treated as potentially coming from an unseen demographic group. This confidence thresholding enables the generation of demographic labels for seen and unseen group membership.

In the second stage, we use the demographic label test samples alongside the seen labeled samples to train a second GCN, whose objective is to classify nodes as either belonging to seen or unseen demographic groups. Once trained, it is applied to the entire dataset—both training and test nodes—to generate a soft label vector $S_s \in [0,1]^{|V|}$. These soft labels S_s are later used in our end-to-end fairness module to ensure fairness for the class prediction. This approach enables the model to incorporate information about demographic uncertainty during training, leading to more robust and equitable classification outcomes across both seen and unseen demographic groups.

3.2 Learning Fair Disentangled Node Embedding

The end-to-end fairness module consists of three main components: a GNN encoder, a class label predictor, and a sensitive attribute classifier. The GNN encoder is responsible for learning two distinct embedding spaces: one for predicting class labels (Z_c) and the other dedicated to predicting sensitive attribute labels (Z_s), using the soft labels established previously. Crucially, our encoder aims not only to derive informative embeddings but also to ensure these representations are independent.

To quantify and enforce this representation independence, we employ a correlation-based measure. Specifically, a correlation matrix $C \in \mathbb{R}^{k \times k}$ is computed, capturing the pairwise correlations among embeddings, where k represents the total dimensionality of embeddings Z_c and Z_s. Given that embeddings are split equally, the first $k/2$ corresponds exclusively to the embeddings for class prediction and the last $k/2$ for seen/unseen group prediction. We isolate the upper-right quadrant of this matrix, denoted $C_{\frac{k}{2},k}$, representing correlations specifically between the class embeddings and sensitive attribute embeddings. This targeted submatrix directly indicates the extent of interdependence between the representations. Our goal is minimize the

maximum correlation between embeddings within this quadrant, which is utilized as our disentanglement loss: $\mathcal{L}_{corr} = \max\{C_{\frac{k}{2},k}\}$.

Following the embedding disentanglement phase, each embedding subspace undergoes further processing through separate fully connected layers. These layers are each followed by a softmax activation function, normalizing predictions into probabilities suitable for classification, as shown in Figure 1(right). Note that the label classifier is only trained with samples from the seen demographic groups. The output predictions ($\hat{y}$ for class labels and $\hat{s}$ for seen/unseen group) are subsequently evaluated using a binary cross-entropy loss function to ensure accurate classification performance: $\mathcal{L}_{ce} = -[y \cdot \log(\hat{p}) + (1-y) \cdot \log(1-\hat{p})]$, where $\hat{p}$ represents either the predicted sensitive attribute probability ($\hat{s}$) or the class label probability ($\hat{y}$).

Besides representation disentanglement, we integrate a fairness regularizer based on statistical parity, which explicitly assesses disparities between seen and unseen sensitive attribute groups. The statistical parity regularizer explicitly quantifies the probability difference of positive predictions across these groups: $\mathcal{L}_{sp} = |p(\hat{y} = 1|s = 0) - p(\hat{y} = 1|s = 1)|$. Here, s indicates the sensitive attribute categorization, distinguishing between seen and unseen attributes, thus ensuring that our model achieves equitable outcomes even for demographic categories absent in training data.

Ultimately, our comprehensive fairness-aware loss function integrates representation disentanglement, classification accuracy, and fairness regularization into a unified training objective: $\mathcal{L}_{final} = \alpha\mathcal{L}_{corr} + (1-\alpha)(\mathcal{L}_{ce}^{class} + \mathcal{L}_{ce}^{s}) + \beta\mathcal{L}_{sp}$. In this formulation, $\mathcal{L}_{ce}^{class}$ is the cross-entropy loss specifically for class label prediction, while $\mathcal{L}_{ce}^{s}$ pertains to sensitive attribute prediction. Hyperparameters α and β are introduced to manage the balance between fairness and predictive accuracy. Specifically, α regulates the intensity of embedding disentanglement, while β controls the contribution of the fairness regularizer. Through careful tuning of these hyperparameters, our model effectively balances fairness constraints with maintaining high predictive utility.

4 Experimental Evaluation

This section presents our experimental results and analysis. The code for our model can be accessed in the following site: https://github.com/frsantosp/FairGRUNT.

4.1 Experimental Setup

We evaluate our method on three real-world datasets: (1) **Tagged** [7] is a spammer detection dataset with 71,128 nodes and 71,226 edges, using age (three groups) as the sensitive attribute; individuals aged 35–50 form the unseen group. (2) **Recidivism** [8] includes 18,877 nodes and 403,978 edges, with the task of predicting bail decisions. The sensitive attribute combines race and age, with black males under 35 and white males over 35 treated as unseen. (3) **Pokec** [9] contains 67,797 nodes and 882,765 edges from a Slovak social network, with the task of predicting sports interest and the sensitive attribute defined by gender and region. Note that each dataset is split into 50% training, 25% validation, and 25% test sets.

We compared FairGRUNT against several baselines, including the standard two-layer **GCN** [10]. **FairGNN** [4] incorporates adversarial debiasing to learn fair

node representations, particularly when some sensitive attribute values are missing. **NIFTY** [5] enforces fairness and stability by introducing random perturbations to node attributes, sensitive features, and graph edges, and maximizes the agreement between predictions on original and perturbed graphs. **BeMap** [6] modifies GNN message passing by sampling neighbors such that the distribution of sensitive attributes is balanced in each node's local neighborhood, promoting fair and unbiased aggregation.

To evaluate the classification performance of the methods, we use the AUC (area under the ROC curve) metric. For fairness assessment, we consider the following two fairness metrics: **Statistical Parity (SP)** and **Equal Opportunity (EO)**.

$$\text{SP} = \left| \max_{s \in X^P} P(\hat{Y} = 1 \mid S = s) - \min_{s \in X^P} P(\hat{Y} = 1 \mid S = s) \right|$$
$$\text{EO} = \left| \max_{s \in X^P} P(\hat{Y} = 1 \mid Y = 1, S = s) - \min_{s \in X^P} P(\hat{Y} = 1 \mid Y = 1, S = s) \right|.$$

Pretraining and Model Setup. The pretraining setup consists of two GCNs, each with two layers and a hidden dimension of 512. As previously mentioned, the first GCN is trained exclusively on samples from seen demographic groups. The second pretraining model is trained using a subset of the test data that contains both seen and unseen groups. The GNN encoder in the end-to-end module also uses a two-layer GCN with a hidden dimension of 512. The output of the GNN encoder has a dimensionality of 4. The label classifier consists of a stack of fully connected layers with a hidden dimension of 512 and is trained only on samples from seen demographic groups. In contrast, the sensitive attribute classifier is trained using samples from both seen and unseen groups and also consists of stacked fully connected layers with a hidden dimension of 512. All neural networks are optimized using Adam with a learning rate of 0.0001. Each model is trained for 1,000 epochs, and all experiments are run using five different random seeds. In our experiments, the hyperparameter α is varied from 0 to 1 in increments of 0.1, while β ranges from 0 to 10. The hyperparameters are chosen based on the model performance on validation set.

4.2 Experimental Results

Table 1 summarizes the results of our experiments. As expected, GCN achieves the highest AUC on all three datasets, since it does not incorporate any fairness constraints and is optimized solely for predictive performance. This establishes an upper bound on classification performance against which fairness-aware methods can be compared. Among the fairness-aware baselines, AUC results vary by dataset. In Pokec, FairGRUNT attains the highest AUC (0.7872), followed closely by FairGNN (0.7751), while Nifty achieves a significantly lower AUC (0.6818). In Tagged, BeMap (0.6521) and Nifty (0.6322) outperform both FairGNN (0.5889) and FairGRUNT (0.5834), indicating stronger utility performance on this dataset. On Bail, Nifty leads among the fairness-aware models with an AUC of 0.9112, followed by BeMap (0.8577), FairGRUNT (0.8365), and FairGNN (0.8313). Notably, FairGRUNT attains comparable AUC scores to other fairness-aware baseline methods, achieving highest AUC scores

Table 1 Performance comparison between the proposed approach and other baseline methods on a) Pokec, b) Tagged and c) Bail datasets.

	AUC	SP	EO
GCN	0.8251±0.0024	0.0626±0.0057	0.0328±0.016
FairGNN	0.7751±0.0592	0.0507±0.0196	0.0241±0.0104
Nifty	0.6818±0.0690	0.0170±0.0356	0.0046±0.0178
BeMap	0.7454±0.0130	0.0571±0.0191	0.1127±0.0229
FairGRUNT	0.7872±0.0012	0.0258±0.0022	0.0260±0.0043

(a) Pokec

	AUC	SP	EO
GCN	0.7196±0.0203	0.1330±0.0165	0.1409±0.0236
FairGNN	0.5889±0.0172	0.0130±0.0290	0.0250±0.0553
Nifty	0.6322±0.0044	0.0382±0.0123	0.0729±0.0247
BeMap	0.6521±0.0172	0.0545±0.0417	0.0473±0.0296
FairGRUNT	0.5834±0.0021	0.0036±0.0019	0.0108±0.0069

(b) Tagged

	AUC	SP	EO
GCN	0.9807±0.0036	0.1914±0.0165	0.1620±0.0236
FairGNN	0.8313±0.1240	0.1007±0.0926	0.1102±0.0857
Nifty	0.9112±0.0372	0.1131±0.0516	0.0998±0.0443
BeMap	0.8577±0.0358	0.2021±0.1006	0.1664±0.0800
FairGRUNT	0.8365±0.1308	0.0380±0.0563	0.0485±0.0854

(c) Bail

on Pokec, but lowest on Tagged. More importantly, FairGRUNT significantly outperforms all baselines on both the Bail and Tagged datasets in terms of fairness, achieving the lowest SP and EO scores. For example, on Tagged, our model obtains SP = 0.0036 and EO = 0.0108, markedly lower than any other method. On Bail, it again leads with SP = 0.0380 and EO = 0.0485, showing that our method effectively mitigates group-level disparities, including the unseen demographic groups.

For the Pokec dataset, Nifty achieves the best fairness metrics (SP = 0.0170, EO = 0.0046), but this comes at the cost of a notably lower AUC. In contrast, FairGRUNT offers a better trade-off between utility and fairness, maintaining a nearly 0.10 higher AUC (0.7872) than Nifty while still achieving competitive fairness scores. The worst fairness performance across all datasets is observed in GCN and BeMap. For GCN, this is expected due to its lack of any fairness objective. BeMap, although relatively strong in AUC, struggles to manage the fairness-utility trade-off effectively, as evidenced by high SP and EO values—especially on the Bail dataset (SP = 0.2021, EO = 0.1664). Overall, these results underscore the strength of our approach in balancing fairness and predictive accuracy on datasets with unseen demographic groups.

5 Conclusions and Future Work

In this work, we present FairGRUNT, a novel fairness-aware GNN model with the inability to handle unseen protected attribute groups. FairGRUNT employs disentangled representation learning to reduce the dependency between sensitive attributes and class labels. Additionally, we incorporated a fairness regularizer based on statistical parity to further reduce discriminatory bias in the learned representations. Our experimental results demonstrate that FairGRUNT effectively improves fairness across several benchmark datasets, while maintaining comparable predictive performance.

For future work, we aim to further improve predictive performance, particularly in settings where the attributes for predicting the class are strongly correlated with those associated with the demographic groups. Alternative strategies beyond feature disentanglement, e.g., using adversarial training, is another research direction.

References

[1] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A Survey on Bias and Fairness in Machine Learning (2022). https://arxiv.org/abs/1908.09635

[2] Rahman, T., Surma, B., Backes, M., Zhang, Y.: Fairwalk: Towards fair graph embedding. Proc of the 28th Int'l Joint Conf on AI, 3289–3295 (2019)

[3] Bose, A.J., Hamilton, W.: Compositional fairness constraints for graph embeddings. arXiv preprint arXiv:1905.10674 (2019)

[4] Dai, E., Wang, S.: Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information. Proc of the 14th ACM Int'l Conf on Web Search and Data Mining, 680–688 (2021)

[5] Agarwal, C., Lakkaraju, H., Zitnik, M.: Towards a unified framework for fair and stable graph representation learning. Uncertainty in AI, 2114–2124 (2021)

[6] Lin, X., Kang, J., Cong, W., Tong, H.: BeMap: Balanced Message Passing for Fair Graph Neural Network (2024). https://arxiv.org/abs/2306.04107

[7] Fakhraei, S., Foulds, J., Shashanka, M., Getoor, L.: Collective spammer detection in evolving multi-relational social networks. In: SIGKDD, pp. 1769–1778 (2015)

[8] Jordan, K., Freiburger, T.: The effect of race/ethnicity on sentencing: Examining sentence type, jail length, and prison length. Journal of Ethnicity in Criminal Justice **13**, 1–18 (2014) https://doi.org/10.1080/15377938.2014.984045

[9] Leskovec, J., Sosič, R.: Snap: A general-purpose network analysis and graph-mining library. ACM TIST **8**(1), 1 (2016)

[10] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)