FairSIN: Achieving Fairness in Graph Neural Networks through Sensitive Information Neutralization

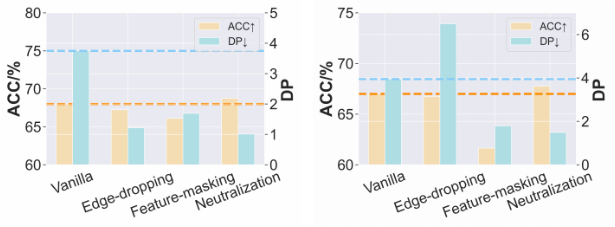
何放

Week6, MAY 2024

Abstract

Bias

GNNs are susceptible to making biased predictions based on sensitive attributes, such as race and gender.



(a) **Pokec-n**

(b) **Pokec-z**

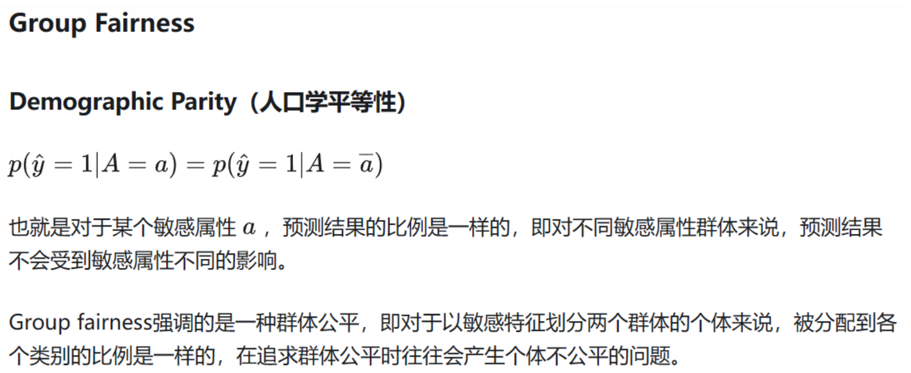
Figure 1: Motivation verification on Pokec datasets. Compared with vanilla GNN without fairness consideration, filtering-based methods, either edge-dropping (Agarwal, Lakkaraju, and Zitnik 2021) or feature-masking (Wang et al. 2022c), always have a trade-off between accuracy (ACC↑) and fairness (DP↓). While our method can improve both.

GNN Bias

The biases in GNN predictions can be attributed to both node features and graph topology

- node features: The raw features of nodes could be statistically correlated to the sensitive attribute, and thus lead to sensitive information leakage in encoded representations.
- graph topology: According to the homophily effects nodes with the same sensitive attribute tend to link with each other, which will make the node representations in the same sensitive group more similar during message passing.

DP



Group Fairness

Demographic Parity（人口学平等性）

$$p(\hat{y}=1|A=a)=p(\hat{y}=1|A=\overline{a})$$

也就是对于某个敏感属性 a ，预测结果的比例是一样的，即对不同敏感属性群体来说，预测结果不会受到敏感属性不同的影响。

Group fairness强调的是一种群体公平，即对于以敏感特征划分两个群体的个体来说，被分配到各个类别的比例是一样的，在追求群体公平时往往会产生个体不公平的问题。

F3

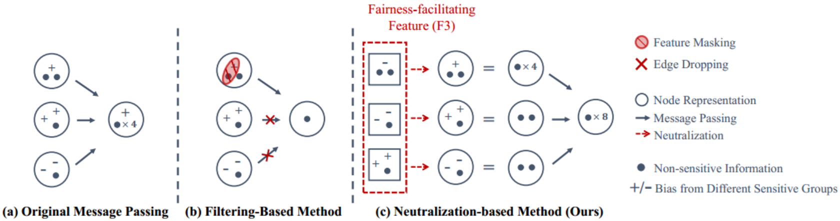
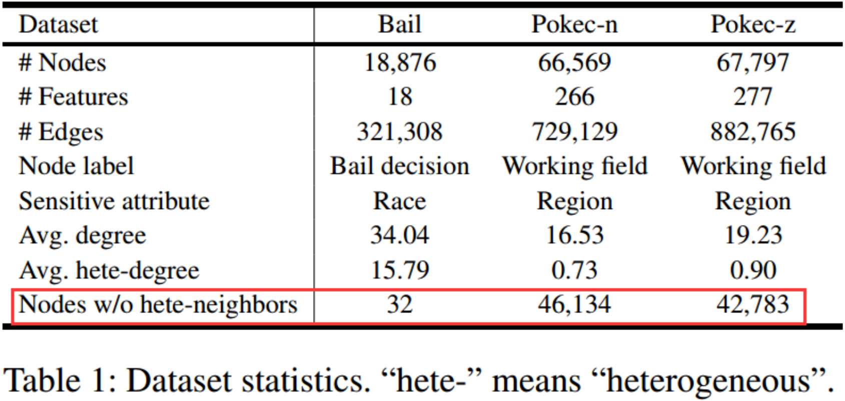

Fairness-facilitating Feature (F3)
+
×4
×8
Feature Masking
Edge Dropping
Node Representation
Message Passing
Neutralization
Non-sensitive Information
+/− Bias from Different Sensitive Groups
(a) Original Message Passing
(b) Filtering-Based Method
(c) Neutralization-based Method (Ours)

Figure 2: Motivation illustration of sensitive information neutralization. Here we assume binary sensitive groups denoted by +/-, and the numbers of +/- indicate the intensity of sensitive information leakage in node representations. (a) Message passing computation will aggregate both non-sensitive feature information (dot symbols) and sensitive biases (+/- symbols); (b) Current SOTA methods are usually *filtering-based* (*e.g.*, edge dropping or feature masking), which may lose much non-sensitive information; (c) Our proposed *neutralization-based* strategy introduces *F3* to statistically neutralize the sensitive bias and provide extra non-sensitive information.

Train an estimator to transfer

Some nodes in real-world graphs have few or even no heterogeneous neighbors, which makes the calculation of F3 infeasible or very uncertain. Therefore, we propose to train an estimator to predict the average features or representations of a node' s heterogeneous neighbors given its own feature.



Dataset	Bail	Pokec-n	Pokec-z
# Nodes	18,876	66,569	67,797
# Features	18	266	277
# Edges	321,308	729,129	882,765
Node label	Bail decision	Working field	Working field
Sensitive attribute	Race	Region	Region
Avg. degree	34.04	16.53	19.23
Avg. hete-degree	15.79	0.73	0.90
Nodes w/o hete-neighbors	32	46,134	42,783

Table 1: Dataset statistics. “hete-” means “heterogeneous”.

Task Definition

semi-supervised node classification task
Graph $\mathcal{G}$ node features X

predict the label $\hat{y}$

labeled node set $\mathcal{V}^L \subset \mathcal{V}$ task1:predict the label $\hat{y} \in \mathcal{Y}$ for every node in the unlabeled node set $\mathcal{V}^U = \mathcal{V} \setminus \mathcal{V}^L$

fairness task

task2:weaken the dependency level between predicted labels $\hat{Y}$ and sensitive attributes S and without losing much classification accuracy

Quantifying Sensitive Information Leakage

$$\mathcal{H}(s|x) = -\mathbb{E}_{(x,s)\sim prior} \log P(s|x),$$

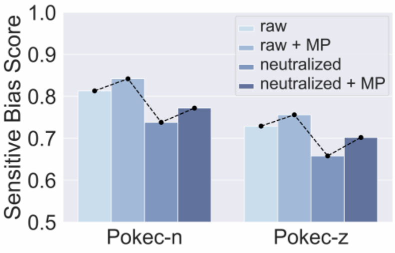
approximate the ground truth predictor
linear intensity function $\mathcal{D}_\theta$

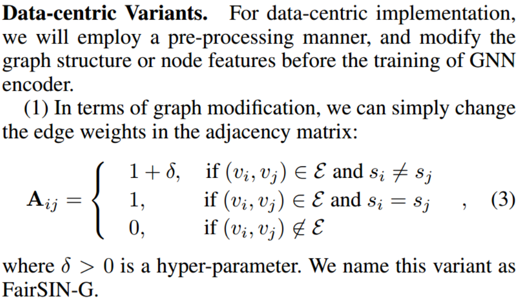
$$\mathcal{D}_\theta(s \mid x) \sim \mathcal{N}(\mu_c, \sigma^2)$$

$$\mathcal{D}_\theta(\bar{s} \mid x) \sim \mathcal{N}(\mu_{ic}, \sigma^2)$$

Normalize the intensity function $\mathcal{D}_\theta$ to define the parameterized predictor $\hat{P}_\theta(s \mid x)$

Message Passing Can Exacerbate Sensitive Biases





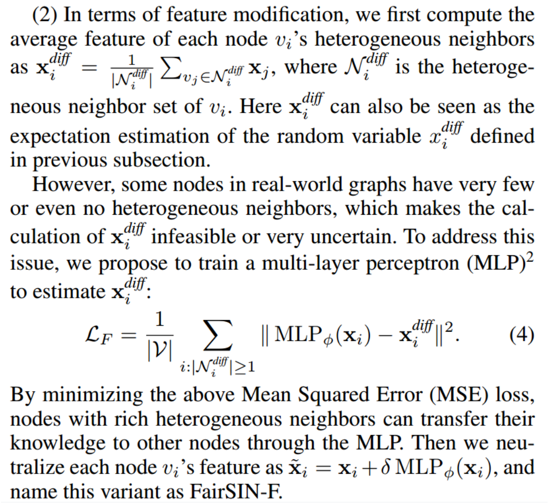
Data-centric Variants. For data-centric implementation, we will employ a pre-processing manner, and modify the graph structure or node features before the training of GNN encoder.

(1) In terms of graph modification, we can simply change the edge weights in the adjacency matrix:

$$\mathbf{A}_{ij} = \begin{cases} 1+\delta, & \text{if } (v_i, v_j) \in \mathcal{E} \text{ and } s_i \neq s_j \\ 1, & \text{if } (v_i, v_j) \in \mathcal{E} \text{ and } s_i = s_j \\ 0, & \text{if } (v_i, v_j) \notin \mathcal{E} \end{cases}, \quad (3)$$

where $\delta > 0$ is a hyper-parameter. We name this variant as FairSIN-G.

Data-centric Variants



(2) In terms of feature modification, we first compute the average feature of each node v_i's heterogeneous neighbors as $\mathbf{x}_i^{\mathit{diff}} = \frac{1}{|\mathcal{N}_i^{\mathit{diff}}|}\sum_{v_j \in \mathcal{N}_i^{\mathit{diff}}} \mathbf{x}_j$, where $\mathcal{N}_i^{\mathit{diff}}$ is the heterogeneous neighbor set of v_i. Here $\mathbf{x}_i^{\mathit{diff}}$ can also be seen as the expectation estimation of the random variable x_i^{diff} defined in previous subsection.

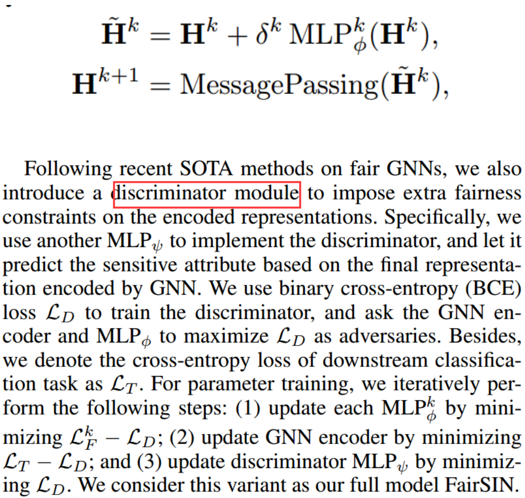
However, some nodes in real-world graphs have very few or even no heterogeneous neighbors, which makes the calculation of $\mathbf{x}_i^{\mathit{diff}}$ infeasible or very uncertain. To address this issue, we propose to train a multi-layer perceptron (MLP)[2] to estimate $\mathbf{x}_i^{\mathit{diff}}$:

$$\mathcal{L}_F = \frac{1}{|\mathcal{V}|} \sum_{i:|\mathcal{N}_i^{\mathit{diff}}| \geq 1} \| \operatorname{MLP}_\phi(\mathbf{x}_i) - \mathbf{x}_i^{\mathit{diff}} \|^2. \qquad (4)$$

By minimizing the above Mean Squared Error (MSE) loss, nodes with rich heterogeneous neighbors can transfer their knowledge to other nodes through the MLP. Then we neutralize each node v_i's feature as $\tilde{\mathbf{x}}_i = \mathbf{x}_i + \delta \operatorname{MLP}_\phi(\mathbf{x}_i)$, and name this variant as FairSIN-F.

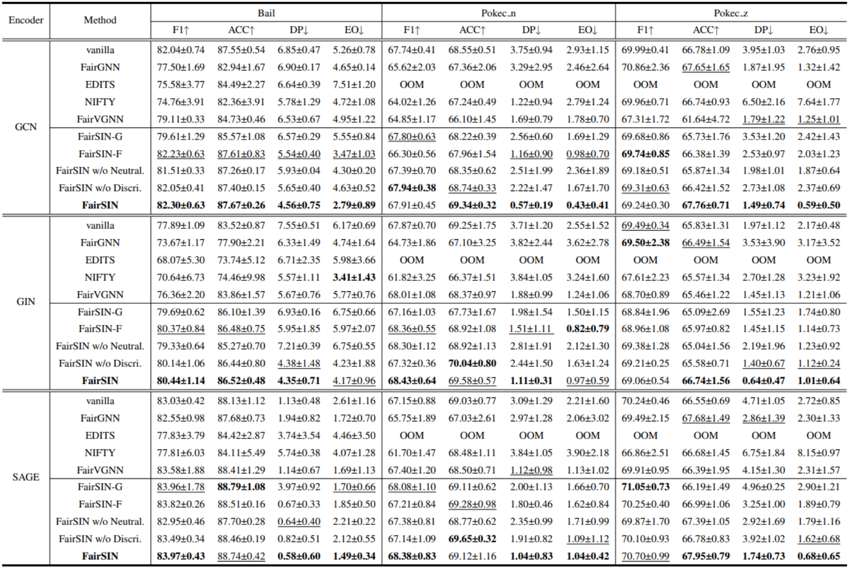
Model-centric Variants

Extends FairSIN-F by jointly learning the MLP_ϕ and GNN encoder.



$$\tilde{\mathbf{H}}^k = \mathbf{H}^k + \delta^k \, \mathrm{MLP}_\phi^k(\mathbf{H}^k),$$
$$\mathbf{H}^{k+1} = \mathrm{MessagePassing}(\tilde{\mathbf{H}}^k),$$

Following recent SOTA methods on fair GNNs, we also introduce a discriminator module to impose extra fairness constraints on the encoded representations. Specifically, we use another MLP_ψ to implement the discriminator, and let it predict the sensitive attribute based on the final representation encoded by GNN. We use binary cross-entropy (BCE) loss $\mathcal{L}_D$ to train the discriminator, and ask the GNN encoder and MLP_ϕ to maximize $\mathcal{L}_D$ as adversaries. Besides, we denote the cross-entropy loss of downstream classification task as $\mathcal{L}_T$. For parameter training, we iteratively perform the following steps: (1) update each MLP_ϕ^k by minimizing $\mathcal{L}_F^k - \mathcal{L}_D$; (2) update GNN encoder by minimizing $\mathcal{L}_T - \mathcal{L}_D$; and (3) update discriminator MLP_ψ by minimizing $\mathcal{L}_D$. We consider this variant as our full model FairSIN.

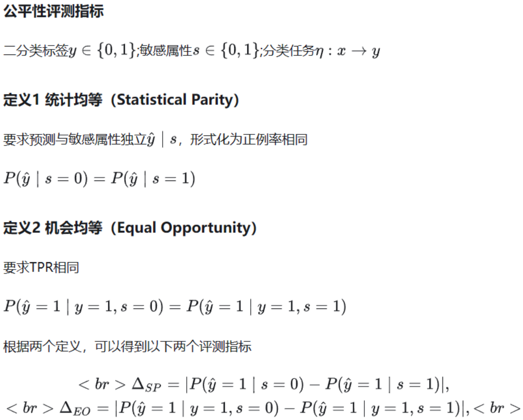


Encoder	Method	Bail				Pokec_n				Pokec_z			
		F1↑	ACC↑	DP↓	EO↓	F1↑	ACC↑	DP↓	EO↓	F1↑	ACC↑	DP↓	EO↓
GCN	vanilla	82.04±0.74	87.55±0.54	6.85±0.47	5.26±0.78	67.74±0.41	68.55±0.51	3.75±0.94	2.93±1.15	69.99±0.41	66.78±1.09	3.95±1.03	2.76±0.95
	FairGNN	77.50±1.69	82.94±1.67	6.90±0.17	4.65±0.14	65.62±2.03	67.36±2.06	3.29±2.95	2.46±2.64	70.86±2.36	67.65±1.65	1.87±1.95	1.32±1.42
	EDITS	75.58±3.77	84.49±2.27	6.64±0.39	7.51±1.20	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	NIFTY	74.76±3.91	82.36±3.91	5.78±1.29	4.72±1.08	64.02±1.26	67.24±0.49	1.22±0.94	2.79±1.24	69.96±0.71	66.74±0.93	6.50±2.16	7.64±1.77
	FairVGNN	79.11±0.33	84.73±0.46	6.53±0.67	4.95±1.22	64.85±1.17	66.10±1.45	1.69±0.79	1.78±0.70	67.31±1.72	61.64±4.72	1.79±1.22	1.25±1.01
	FairSIN-G	79.61±1.29	85.57±1.08	6.57±0.29	5.55±0.84	67.80±0.63	68.22±0.39	2.56±0.60	1.69±1.29	69.68±0.86	65.73±1.76	3.53±1.20	2.42±1.43
	FairSIN-F	82.23±0.63	87.61±0.83	5.54±0.40	3.47±1.03	66.30±0.56	67.96±1.54	1.16±0.90	0.98±0.70	**69.74±0.85**	66.38±1.39	2.53±0.97	2.03±1.23
	FairSIN w/o Neutral.	81.51±0.33	87.26±0.17	5.93±0.04	4.30±0.20	67.39±0.70	68.35±0.62	2.51±1.99	2.36±1.89	69.18±0.51	65.87±1.34	1.98±1.01	1.87±0.64
	FairSIN w/o Discri.	82.05±0.41	87.40±0.15	5.65±0.40	4.63±0.52	**67.94±0.38**	68.74±0.33	2.22±1.47	1.67±1.70	69.31±0.63	66.42±1.52	2.73±1.08	2.37±0.69
	FairSIN	**82.30±0.63**	**87.67±0.26**	**4.56±0.75**	**2.79±0.89**	67.91±0.45	**69.34±0.32**	**0.57±0.19**	**0.43±0.41**	69.24±0.30	**67.76±0.71**	**1.49±0.74**	**0.59±0.50**
GIN	vanilla	77.89±1.09	83.52±0.87	7.55±0.51	6.17±0.69	67.87±0.70	69.25±1.75	3.71±1.20	2.55±1.52	69.49±0.34	65.83±1.31	1.97±1.12	2.17±0.48
	FairGNN	73.67±1.17	77.90±2.21	6.33±1.49	4.74±1.64	64.73±1.86	67.10±3.25	3.82±2.44	3.62±2.78	**69.50±2.38**	66.49±1.54	3.53±3.90	3.17±3.52
	EDITS	68.07±5.30	73.74±5.12	6.71±2.35	5.98±3.66	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	NIFTY	70.64±6.73	74.46±9.98	5.57±1.11	**3.41±1.43**	61.82±3.25	66.37±1.51	3.84±1.05	3.24±1.60	67.61±2.23	65.57±1.34	2.70±1.28	3.23±1.92
	FairVGNN	76.36±2.20	83.86±1.57	5.67±0.76	5.77±0.76	68.01±1.08	68.37±0.97	1.88±0.99	1.24±1.06	68.70±0.89	65.46±1.22	1.45±1.13	1.21±1.06
	FairSIN-G	79.69±0.62	86.10±1.39	6.93±0.16	6.75±0.66	67.16±1.03	67.73±1.67	1.98±1.54	1.50±1.15	68.84±1.96	65.09±2.69	1.55±1.23	1.74±0.80
	FairSIN-F	80.37±0.84	86.48±0.75	5.95±1.85	5.97±2.07	68.36±0.55	68.92±1.08	1.51±1.11	**0.82±0.79**	68.96±1.08	65.97±0.82	1.45±1.15	1.14±0.73
	FairSIN w/o Neutral.	79.33±0.64	85.27±0.70	7.21±0.39	6.75±0.55	68.30±1.12	68.92±1.13	2.81±1.91	2.12±1.30	69.38±1.28	65.04±1.56	2.19±1.96	1.23±0.92
	FairSIN w/o Discri.	80.14±1.06	86.44±0.80	4.38±1.48	4.23±1.88	67.32±0.36	**70.04±0.80**	2.44±1.50	1.63±1.24	69.21±0.25	65.58±0.71	1.40±0.67	1.12±0.24
	FairSIN	**80.44±1.14**	**86.52±0.48**	**4.35±0.71**	4.17±0.96	**68.43±0.64**	69.58±0.57	**1.11±0.31**	0.97±0.59	69.06±0.54	**66.74±1.56**	**0.64±0.47**	**1.01±0.64**
SAGE	vanilla	83.03±0.42	88.13±1.12	1.13±0.48	2.61±1.16	67.15±0.88	69.03±0.77	3.09±1.29	2.21±1.60	70.24±0.46	66.55±0.69	4.71±1.05	2.72±0.85
	FairGNN	82.55±0.98	87.68±0.73	1.94±0.82	1.72±0.70	65.75±1.89	67.03±2.61	2.97±1.28	2.06±3.02	69.49±2.15	67.68±1.49	2.86±1.39	2.30±1.33
	EDITS	77.83±3.79	84.42±2.87	3.74±3.54	4.46±3.50	OOM	OOM	OOM	OOM	OOM	OOM	OOM	OOM
	NIFTY	77.81±6.03	84.11±5.49	5.74±0.38	4.07±1.28	61.70±1.47	68.48±1.11	3.84±1.05	3.90±2.18	66.86±2.51	66.68±1.45	6.75±1.84	8.15±0.97
	FairVGNN	83.58±1.88	88.41±1.29	1.14±0.67	1.69±1.13	67.40±1.20	68.50±0.71	1.12±0.98	1.13±1.02	69.91±0.95	66.39±1.95	4.15±1.30	2.31±1.57
	FairSIN-G	83.96±1.78	**88.79±1.08**	3.97±0.92	1.70±0.66	68.08±1.10	69.11±0.62	2.00±1.13	1.66±0.70	**71.05±0.73**	66.19±1.49	4.96±0.25	2.90±1.21
	FairSIN-F	83.82±0.26	88.51±0.16	0.67±0.33	1.85±0.50	67.21±0.84	69.28±0.98	1.80±0.46	1.62±0.84	70.25±0.40	66.99±1.06	3.25±1.00	1.89±0.79
	FairSIN w/o Neutral.	82.95±0.46	87.70±0.28	0.64±0.40	2.21±0.22	67.38±0.81	68.77±0.62	2.35±0.99	1.71±0.99	69.87±1.70	67.39±1.05	2.92±1.69	1.79±1.16
	FairSIN w/o Discri.	83.49±0.34	88.46±0.19	0.82±0.51	2.12±0.55	67.14±1.09	**69.65±0.32**	1.91±0.82	1.09±1.12	70.10±0.93	66.78±0.83	3.92±1.02	1.62±0.68
	FairSIN	**83.97±0.43**	88.74±0.42	**0.58±0.60**	**1.49±0.34**	**68.38±0.83**	69.12±1.16	**1.04±0.83**	**1.04±0.42**	70.70±0.99	**67.95±0.79**	**1.74±0.73**	**0.68±0.65**

Table 2: Comparison among SOTA methods and different variants of FairSIN. (Bold: the best; underline: the runner-up.)

FairGNN (WSDM’21)

数据集Pokec：包含社交网络中的匿名数据，用户画像包括年龄、性别、爱好、兴趣、教育、工作领域等。
按照用户所属省份不同，划分数据集为 Pokec-z和 Pokec-n 将用户所属省份作为敏感属性， 分类任务 是预测用户的工作领域



公平性评测指标

二分类标签$y \in \{0,1\}$;敏感属性$s \in \{0,1\}$;分类任务$\eta : x \rightarrow y$

定义1 统计均等（Statistical Parity）

要求预测与敏感属性独立$\hat{y} \mid s$，形式化为正例率相同

$$P(\hat{y} \mid s=0) = P(\hat{y} \mid s=1)$$

定义2 机会均等（Equal Opportunity）

要求TPR相同

$$P(\hat{y}=1 \mid y=1, s=0) = P(\hat{y}=1 \mid y=1, s=1)$$

根据两个定义，可以得到以下两个评测指标

$$< br > \Delta_{SP} = |P(\hat{y}=1 \mid s=0) - P(\hat{y}=1 \mid s=1)|,$$
$$< br > \Delta_{EO} = |P(\hat{y}=1 \mid y=1, s=0) - P(\hat{y}=1 \mid y=1, s=1)|, < br >$$

RQs

- RQ1: How effective is our proposed method compared with SOTA graph fairness methods?
- RQ2: How does each module of our proposed method contribute to the final performance?
- RQ3: How does the hyper-parameter δ influence the performance?
- RQ4: How does the time cost of our method compared with other baselines?

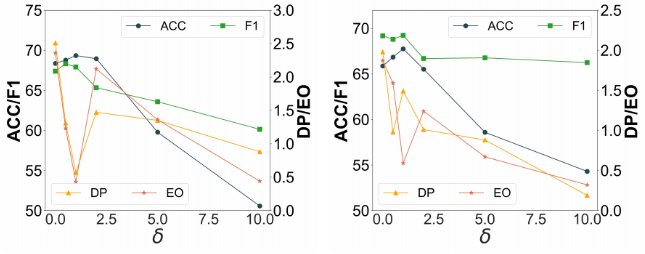
FairSIN-F VS FairSIN-G

Consequently, in comparison, the predictive performance of FairSIN-G falls short when contrasted with FairSIN-F.
It is worth noting that as a pre-processing method,
FairSIN-F is only slightly worse than the model-centric variant FairSIN,and outperforms previous SOTA methods.
Therefore, FairSIN-F offers a cost-effective, model-agnostic and task-irrelevant solution for fair node representation learning.

Ablated Models

FairSIN without Discri. denotes the version of FairSIN without the discriminator.
FairSIN without Neutral. denotes the version ofFairSIN where $\delta = 0$.

Hyper-parameter Analysis

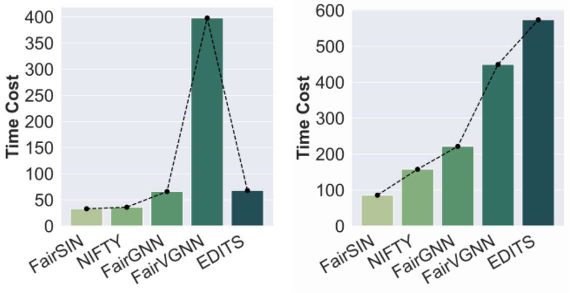


(a) Pokec-n

(b) Pokec-z

Figure 4: Classification performance and group fairness under different values of hyper-parameter δ.

Efficiency Analysis



(a) Bail

(b) Credit

Figure 5: Training time cost on Bail and Credit with GCN backbone (in seconds).

We can find that FairSIN has the lowest time cost among all methods. Thus our method is both efficient and effective, enabling potential applications in various scenarios.

Conclusion

Summary

By emphasizing the features of each node' s heterogeneous neighbors, F3 can simultaneously neutralize the sensitive bias in node representations and provide extra non-sensitive feature information.

Future task

Besides, current F3 are irrelevant to downstream tasks, and it is also possible to build task-specific ones.

In addition, when we need to handle multiple sensitive groups at the same time, we can extend F3 to neutralize a joint distribution of sensitive attributes.

Some ideas

idea1

《Say No to the Discrimination: Learning Fair Graph Neural Networks with Limited Sensitive Attribute Information》
《Towards rawlsian difference principle on graph convolutional network》——ACM 《Uncovering the Structural Fairness in Graph Contrastive Learning》——Advances in neural information processing systems

idea2

bias和fairness的定义？二元敏感属性抵消，符号边也是根据正负的奇偶来判断敌友在link prediction里定义类似于敏感属性的东西？比如：符号边的正负强度（？