Fairness in Graph Representation Learning with Regularization

Sharmishtha Dutta
RCS: duttas, RIN: 661990701

Graduate Student

Department of Computer Science

Rensselaer Polytechnic Institute

September 24, 2024

Background: Fairness in Graph

FairDrop [1] and FairAdj[2] are the two recent works with competitive results on the tasks of link prediction and node classification of graph datasets.

- FairDrop[1] is a biased edge dropout method to enhance fairness.
- FairAdj[2] learns a fair adjacency matrix for link prediction task. The model takes a variational approach and employs two different optimization processes, one for learning a fair version of the adjacency matrix and one for the link prediction.
- FairGo[3] is a model-agnostic approach that uses filters to remove sensitive attributes from learned embeddings. In [4], the authors remove sensitive attributes and modify the structural edge distribution of the input graph by only conditioning on non-sensitive attributes. The paper explores two directions, weighting, and regularization, to achieve this.

Algorithm Idea

- Many existing works on fair learning in graphs rely on dropout methods or explicitly removing sensitive attributes before or after learning.

Regularization

We propose to use regularization to limit the mutual information between the learned representation and the sensitive attribute(s).

Mutual information (MI)

Measures for Mutual Information

Let $\kappa : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ be a real valued positive kernel that is also infinitely divisible. Given $\{x_i\}_{i=1}^{n} \subset \mathcal{X}$, each x_i being a real-valued scalar or vector, and the Gram matrix K obtained from $K_{ij} = \kappa(x_i, x_j)$, a matrix-based analogue to Renyi's α-order entropy can be defined as:

$$S_\alpha(G) = \frac{1}{1-\alpha} \log_2 \left(tr(G^\alpha) \right)$$

where, $G = \frac{1}{n} \frac{G_{ij}}{\sqrt{G_{ii} G_{jj}}}$ is the normalized kernel matrix.
Renyi's α-order mutual information can be derived as:

$$I_\alpha(G^i, G^j) = S_\alpha(G^i) + S_\alpha(G^j) - S_\alpha(G^i, G^j)$$

where, the joint entropy is:

$$S_\alpha(G^i, G^j) = S_\alpha\left(\frac{G^i \circ G^j}{tr(G^i \circ G^j)}\right)$$

Algorithm

Task

Representation learning in graph datasets (e.g. DBLP, Cora) [assume, only one sensitive attribute]

Closely following the experimental setup on [1], we follow these steps for link prediction:

1. Use Graph convolutional networks (GCN)[6] to learn node representations
2. Augment the loss function of GCN with the mutual information between the learned representation and the sensitive attribute

$$\mathcal{L} = L_{GCN}(X, A) + \lambda \mathcal{I}(S, Z)$$

Here,
L_{GCN} is the loss for GCN as a function of the node attribute, X, and adjacency information, A
$\lambda \in [0, 1]$ is a regularization hyperparameter
$\mathcal{I}(S, Z)$ is the mutual information between the learned representation (output of GCN), Z, and the sensitive attribute, S

Idea

Use the adjacency matrix and feature matrix to generate node embeddings.

$$Z = (H^{(L)})$$

where, L is the total number of layers.

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{\frac{1}{2}}H^{(l)}W^{(l)})$$

here, $\tilde{A} = A + \mathcal{I}_N$ is the adjacency matrix with self connections
$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ is the degree matrix
$W^{(l)}$ is the layer specific trainable parameter
$H^{(0)} = \mathbf{X}$ is the initial encodings

Evaluation

Benchmarks to compare

1. FairDrop[1]: To measure performance of dropout based vs regularization based approaches

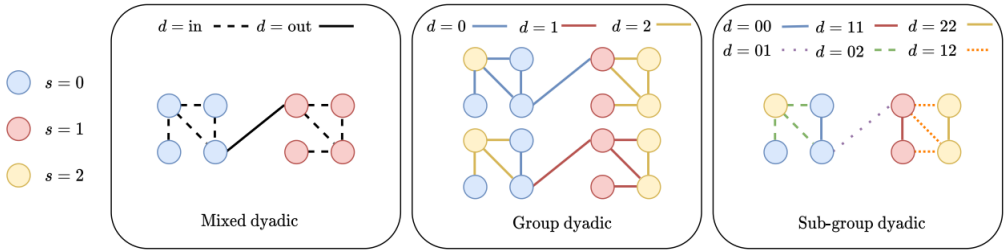
Dataset

Cora and Citeseer: citation network, the paper classes treated as sensitive attribute

Metrics for evaluation

- Measure the link prediction quality with accuracy and the area under the ROC curve (AUC)
- Measure fairness of the predictions, use the demographic parity difference and the equality of odds difference using the dyadic groups described in [1]

Metrics



1. Mixed dyadic: $|D| = 2$
2. Group dyadic: $|D| = |S|$
3. Sub-group dyadic: $|D| = \frac{(|S|+2-1)!}{2!(|S|-1)!}$

Demographic parity for mixed dyadic, where D=0 denotes inter-group links and D=1 denotes intra-group links:

$$\mathbb{P}(\hat{Y}|D = 0) = \mathbb{P}(\hat{Y}|D = 1)$$

Results on Cora

Model	ACC ↑	AUC ↑	$\Delta DP_m \downarrow$	$\Delta EO_m \downarrow$	$\Delta DP_g \downarrow$	$\Delta EO_g \downarrow$
Fairdrop	**82.4 ± 0.9**	**90.1 ± 0.7**	52.9 ± 2.5	31.0±4.9	**11.8±3.2**	**14.9±3.7**
Ours	63.50 ± 0.76	67.88 ± 1.38	**11.87 ± 1.53**	**14.26±3.37**	23.50±2.33	33.61±4.69

Table: Accuracy, Area under the ROC curve, Demographic parity difference (ΔDP), Equalized odds difference (ΔEO) in Cora dataset

Model	$DP_s \downarrow$	$\Delta EO_s \downarrow$
Fairdrop	89.4 ± 3.4	100
Ours	**54.82 ± 7.24**	100

Table: Demographic parity difference (ΔDP), Equalized odds difference (ΔEO) in Cora dataset

Results on Citeseer

Model	ACC ↑	AUC ↑	$\Delta DP_m \downarrow$	$\Delta EO_m \downarrow$	$\Delta DP_g \downarrow$	$\Delta EO_g \downarrow$
Fairdrop	**79.2 ± 1.4**	**88.4 ± 1.4**	42.6 ± 2.5	26.5±4.2	**18.7±4.0**	**17.6±5.5**
Ours	64.24 ± 0.83	64.88 ± 0.22	**20.48 ±0.57**	**8.13±1.75**	47.51±4.57	67.23±3.53

Table: Accuracy, Area under the ROC curve, Demographic parity difference (ΔDP), Equalized odds difference (ΔEO) in Citeseer dataset

Model	$DP_s \downarrow$	$\Delta EO_s \downarrow$
Fairdrop	**55.7 ± 1.5**	**26.6 ± 2.6**
Ours	76.26 ± 5.66	83.81 ± 6.2

Table: Demographic parity difference (ΔDP), Equalized odds difference (ΔEO) in Citeseer dataset

References

1. Spinelli, Indro, et al. "FairDrop: Biased Edge Dropout for Enhancing Fairness in Graph Representation Learning." IEEE Transactions on Artificial Intelligence (2021).
2. Li, Peizhao, et al. "On dyadic fairness: Exploring and mitigating bias in graph connections." International Conference on Learning Representations. 2020.
3. Wu, Le, et al. "Learning fair representations for recommendation: A graph-based perspective." Proceedings of the Web Conference 2021. 2021.
4. Wang, Nan, et al. "Unbiased Graph Embedding with Biased Graph Observations." arXiv preprint arXiv:2110.13957 (2021).
5. Gong, Tieliang, et al. "Computationally Efficient Approximations for Matrix-based Renyi's Entropy." arXiv preprint arXiv:2112.13720 (2021).
6. Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).