Adversarial Inter-Group Link Injection Degrades the Fairness of Graph Neural Networks

Hussain Hussain*  
Graz University of Technology  
Know-Center GmbH  
hussain@tugraz.at

Meng Cao*  
Nanjing University  
camoeng@smail.nju.edu.cn

Sandipan Sikdar  
L3S Research Center  
sandipan.sikdar@l3s.de

Denis Helic  
Modul University Vienna  
denis.helic@modul.ac.at

Elisabeth Lex  
Graz University of Technology  
elisabeth.lex@tugraz.at

Markus Strohmaier  
University of Mannheim  
markus.strohmaier@uni-mannheim.de

Roman Kern  
Graz University of Technology  
rkern@tugraz.at

Graz University of Technology
elisabeth.lex@tugraz.at

Graz University of Technology
markus.strohmaier@uni-mannheim.de

Graz University of Technology
rkern@tugraz.at

Graz University of Technology
know-center.gmbh

**Abstract**—We present evidence for the existence and effectiveness of adversarial attacks on graph neural networks (GNNs) that aim to degrade fairness. These attacks can disadvantage a particular subgroup of nodes in GNN-based node classification, where nodes of the underlying network have sensitive attributes, such as race or gender. We conduct qualitative and experimental analyses explaining how adversarial link injection impairs the fairness of GNN predictions. For example, an attacker can compromise the fairness of GNN-based node classification by injecting adversarial links between nodes belonging to opposite subgroups and opposite class labels. Our experiments on empirical datasets demonstrate that adversarial fairness attacks can significantly degrade the fairness of GNN predictions (attacks are effective) with a low perturbation rate (attacks are efficient) and without a significant drop in accuracy (attacks are deceptive). This work demonstrates the vulnerability of GNN models to adversarial fairness attacks. We hope our findings raise awareness about this issue in our community and lay a foundation for the future development of GNN models that are more robust to such attacks.

**Index Terms**—fairness, adversarial attacks, graph neural networks

I. INTRODUCTION

Graph neural networks (GNNs) are susceptible to adversarial attacks targeting prediction accuracy [5], [11], [29]. Recent studies have also uncovered possibilities for adversarial attacks to compromise the fairness of classical machine learning models [15]. However, we still lack a deeper insight into GNNs’ susceptibility to adversarial attacks on prediction fairness as well. Here, we think of prediction fairness as a similar or equal treatment of (social) subgroups for a given prediction task. For example, in the job-hunting market where candidates are socially connected, employers use GNNs to decide the salary of each candidate. Attackers from a specific demographic group, e.g. a certain race, gender, or religion, can intentionally conduct adversarial attacks, with the aim of making the model more likely to predict high salary for their own group, while severely hurting the benefit of the other groups (cf. Fig. 1). Hence, adversarial attacks on the fairness of GNNs enable an attacker to deliberately put individuals belonging to a subgroup of nodes at a disadvantage. We thus turn our attention to the following problem:

**Problem.** We define and analyze adversarial attack strategies...
that degrade the fairness of GNN-based node classifiers.

**Method.** We base our research on two streams of research: adversarial attacks on GNNs that aim to reduce GNN classification accuracy [5], [13], [22], [24], [29], [30], and attacks on fairness in the context of classical machine learning [3], [16]–[18]. We measure fairness in terms of statistical parity, which captures the level of independence between predictions and group memberships. In our node classification problem, we assume a binary class label $y$ with negative ($y=0$) or positive ($y=1$) values. Each node also has a binary sensitive attribute $s$ representing its group: privileged ($s=0$) or unprivileged ($s=1$).

First, we formally define the problem of Fairness Attacks on GNNs (FA-GNN). We then present four adversarial linking strategies based on injecting adversarial links between two subsets of nodes (from a specific class label and sensitive attribute value pair). Through qualitative analysis, we develop insights on the influence of these linking strategies on the statistical parity of GNN predictions. We further illustrate these strategies on synthetically generated graphs by evaluating the prediction fairness of the graph convolution network (GCN) model [12]. Finally, we evaluate FA-GNN adversarial linking strategies on three real-world social network datasets. Fig. 1, illustrates an FA-GNN strategy on the NBA player social network [4], where the attack degrades the fairness of GNN predictions.

**Results.** Our qualitative analysis and experimental evaluation demonstrate the existence and success of adversarial attacks on the fairness of GNN-based classifiers. In particular, our results suggest that adding edges between nodes belonging to opposite groups and opposite class labels leads to less fair predictions. This finding is interesting since it appears plausible that intra-group links promote fairness [14], [19].

**Contributions and implications.** First, we present the problem of adversarial attacks on GNNs that can impair the fairness of node classification. Second, we provide theoretical insights on the consequences of adversarial link injection on the fairness of GNN-based node classifiers. Third, through experiments on three real-world datasets, we demonstrate the effectiveness, efficiency, and deceptiveness of fairness attacks. Our work raises awareness about the vulnerability of GNNs to fairness attacks and argues for the development of models that are more robust against such attacks.

**II. PRELIMINARIES**

**Notation.** We consider a graph $G(V,E,X,y,s)$, where $V = \{1, \ldots, N\}$ is the set of nodes, $E \subseteq V \times V$ is the set of edges, $X \in \mathbb{R}^{N \times F}$ is the feature matrix, while $y \in \{0,1\}^N$ represent the node labels and $s \in \{0,1\}^N$ represent the node sensitive attributes. Hence, each node $u \in V$ has a label $y(u)$ denoting its class, a sensitive attribute (e.g., race, gender, etc.) $s(u)$, whose value determines its group membership, and an $F$-dimensional feature vector $X(u)$.

**Subsets.** We consider a partition $S$ of the node set $V$ into four subsets: $S = \{y_0 s_0, y_0 s_1, y_1 s_0, y_1 s_1\}$, where $y_i s_j$ denotes the subset of nodes with value pair $y(.) = i$ and $s(.) = j$, with $n_{y,s}$ representing its cardinality. We overload this notation to represent the number of nodes in a group $s = j$ as $n_s$. Hereafter, the term subset denotes nodes with a particular combination of label and sensitive attribute values, unlike the term group that denotes nodes with a particular sensitive attribute value.

**Graph neural networks for node classification.** GNNs [25] are machine learning models that operate on graph data. GNNs update a node’s representation by aggregating the initial representations of its neighbors. In (semi-)supervised node classification, we train a GNN on the ground truth classes of labeled nodes $V_l \subseteq V$. For binary node classification, the GNN outputs the predicted probability of each node belonging to class $y = 1$, i.e., $GNN : G(V,E,X) \rightarrow [0,1]^N$. After thresholding, this results in one prediction per node $\hat{y}(u) \in \{0,1\}$.

**Structural adversarial attacks on GNNs.** Given a clean graph $G(V,E,X,y,s)$, an adversarial structural attack [11] modifies this graph and structure perturbed graph $G'$ by perturbing the clean edge set. The number of allowed modifications on the edge set is bounded by a perturbation rate $\delta$, i.e., the attacker can add or remove up to $\delta |E|$ edges. We consider that the attacker modifies the graph prior to the GNN training and evaluation (poisoning attack). Such adversarial attacks increase the prediction error of the GNN on the perturbed graph, particularly the error on nodes directly targeted by the attack [29]. The GNN model trained on the $G'$ is called the victim GNN model.

**Fairness metrics.** We measure the performance of FA-GNN linking strategies in terms of the statistical parity difference $SPD$ [6], that measures the difference in predictive outcomes between the two groups. To recognize which group is at a disadvantage, we use a signed difference.

$$SPD = P(\hat{y} = 1|s = 0) - P(\hat{y} = 1|s = 1).$$

**III. FAIRNESS ATTACKS ON GNNs**

In this section, we devise, analyze and illustrate fairness attacks on GNNs via adversarial link injection.

**A. Problem formulation**

**Attacker’s goal.** The attacker’s goal is to degrade the prediction fairness of a victim GNN binary node classifier by increasing the absolute statistical parity difference $|SPD|$. 

**Attacker’s knowledge.** The attacker has access to the graph structure, node features and sensitive attributes as well as the ground-truth labels of some nodes $W \subseteq V$, that is $(V,E,X,y(W),s)$. For example, in the job-hunting market, an attacker may have access to the candidates’ relationships in the social network and their attributes and demographics, but can only obtain the salary of some candidates. The attacker has no access to the parameters or hyperparameters of the victim GNN model. However, similar to [29], [30], the attack uses a surrogate GNN model to predict labels for the unlabeled nodes, which makes the attack a grey-box attack [11].
denote the predictions of the surrogate GCN as \( \hat{y} \) to distinguish them from the predictions of the victim GNN \( y \).

**Attacker’s capabilities.** The attacker is able to inject edges into the graph up to a budget, defined by a perturbation rate \( \delta > 0 \), i.e., the attacker can add up to \( \delta |E| \) edges to the graph. Note that we demonstrate fairness attacks through link injection and leave other possible methods for future work.

**B. Attack strategy**

We now introduce adversarial linking strategies for fairness attacks on GNNs (FA-GNN) that work within our settings. These linking strategies take a clean graph \( G(V, E, X, y(W), s) \) as an input and adds edges between random nodes from subset \( A \in \mathcal{S} \) and random nodes from subset \( B \in \mathcal{S} \) up to a perturbation rate \( \delta \). To be able to partition the graph, the attacker first trains a surrogate GCN model to predict the labels for unlabeled nodes \( \hat{y}(V \setminus W) \).

The attacker then sets the predictions of labeled nodes as the known ground-truth labels \( \hat{y}(W) := y(W) \). This provides an approximation of which nodes belong to \( A \) and \( B \). Finally, the attacker adds \( \lfloor \delta |E| \rfloor \) edges between random nodes from \( A \) and \( B \).

Choosing \( A, B \) reduces to one of the following four linking strategies (illustrated in Fig. 2 - Top) that connect nodes of

1. **DD**: Different class and Different group,
2. **DE**: Different class and Equal group,
3. **ED**: Equal class and Different group, or
4. **EE**: Equal class and Equal group, that is \( A = B \).

Hereafter, we mainly use the term FA-GNN to refer to the aforementioned linking strategies.

**C. Attack consequences**

We investigate the effects of the different adversarial linking strategies on fairness in terms of statistical parity difference (SPD). As SPD captures the predictive difference between the groups, we study the attack effects in terms of the classification errors for those groups, that is, the false positives and false negatives in each group. First, we rewrite Equation 1 in terms of our four subsets:

\[
SPD = P(\hat{y} = 1, y = 0 | s = 0) + P(\hat{y} = 1, y = 1 | s = 0) - P(\hat{y} = 1, y = 0 | s = 1) - P(\hat{y} = 1, y = 1 | s = 1).
\]

(2)

We rewrite with the error terms:

\[
SPD = \frac{FP_{s_0} - FN_{s_0}}{n_{s_0}} - \frac{FP_{s_1} - FN_{s_1}}{n_{s_1}} + \frac{n_{y_1s_0}}{n_{s_0}} - \frac{n_{y_1s_1}}{n_{s_1}},
\]

(3)

where \( FN_{s_j} \) and \( FP_{s_j} \) are the false negative and false positive counts in group \( s = j \). If we consider a random classifier or a constant classifier, the terms in Equation 3 cancel out and \( SPD = 0 \). In case of a perfect classifier, all error terms vanish in Equation 3, and we get \( SPD = n_{y_1s_0}/n_{s_0} - n_{y_1s_1}/n_{s_1} \). Hence, a perfect predictor is as fair as the original label distribution.

We use Equation 3 to investigate the different linking strategies of FA-GNN. We explore homophilic graphs in our analysis (where likes attract), as GNNs are typically deployed on homophilic graphs [28]. According to previous work [27], adding edges between nodes of different classes increases the error rates on these nodes. Conversely, adding edges between nodes of the same class decreases the error rates on these nodes. We neglect the changes of errors on nodes that are not involved in the linking, since directly attacked nodes usually exhibit a higher error [29]. We further address this assumption and perform a similar analysis on heterophilic graphs in the supplemental information\(^2\). After performing the attack on the clean graph \( G \), we get a new statistical parity difference on the perturbed graph \( G' \), that is \( SPD' = SPD + \Delta SPD \). We denote the change in the number of false positives and false negatives on \( s_j \) as \( \Delta FP_{s_j} \) and \( \Delta FN_{s_j} \), respectively.

**Analysis.** Without loss of generality, we discuss the adversarial linking strategies from the perspective of nodes in \( A = y_1s_1 \).

**Case DD:** \( B = y_0s_0 \). Here, we link nodes from different classes, so error rates would increase:

\[
\Delta FP_{s_0} \geq 0, \Delta FN_{s_1} \geq 0, \Delta SPD = \frac{\Delta FP_{s_0}}{n_{s_0}} + \frac{\Delta FN_{s_1}}{n_{s_1}} \geq 0
\]

Therefore, this linking strategy results in an attack against group \( s_1 \). Note that we neglect \( \Delta FP_{s_1} \) and \( \Delta FN_{s_0} \) since they are not directly affected by the attack. By applying the same logic to the remaining cases, we find that:

**Case DE:** \( B = y_0s_1 \). This linking strategy results in a change on \( SPD \) but the direction is not predictable, so the attack is not targeted, that is, it does not specifically increase or decrease \( SPD \).

\(^2\)We provide supplemental information with detailed analysis and extended evaluation at https://arxiv.org/abs/2209.05957.
CaseED: $B = y_1s_0$. The attack is also not targeted.  
CaseEE: $B = y_1s_1$. This linking strategy results in an attack in favor of group $s_1$.

In summary, linking $y_1s_1$ and $y_0s_0$ (nodes from different classes and different groups, that is DD) results in an attack targeted against $s_1$. Besides, increasing the intra-connectivity of $y_1s_1$ (that is, EE) results in a targeted attack against $s_0$. EE only manipulates one subset, which might not be as effective as DD that increases the connectivity between two subsets.

D. Illustration on synthetic graphs

We illustrate FA-GNN linking strategies on synthetic graphs and observe the GNN’s response in terms of $SPD$. We consider graphs of 4000 nodes divided equally into four subsets of size 1000. We later illustrate the strategies on empirical datasets, where the groups are unbalanced.

Generating the clean graph. We assign 10 features for each node, resulting in a feature matrix $X \in \mathbb{R}^{4000 \times 10}$, where $X(u, k)$ is the $k$-th feature of node $u$. For a node $u \in V$, we sample 10 features from a normal distribution $X(u, k) \sim \mathcal{N}(\mu_y, (k), \sigma^2)$ where $1 \leq k \leq 10$. $\mu_y \in [-1, +1]$ is the mean vector sampled independently at random from $[-1, +1]$, and $\sigma^2 = 0.5$. To build the graphs, we use the stochastic block model [9] (SBM) on the four subsets. As classical GNNs are typically deployed on homophilic graphs [10], [28], we follow a setup where the clean graph is homophilic w.r.t. labels.

Perturbing the clean graph. We apply the four strategies DD, DE, ED, EE on subset $A = y_1s_1$. For a perturbation rate $\delta$, we add $\delta |E|$ edges between random nodes from $A$ and random nodes from $B \in \mathcal{S}$. Note that in these illustrative experiments, all labels are available to the attacker, so the attacker does not utilize a surrogate GCN.

Evaluation. After building the graph, we train a (GCN) model for node classification. For the training process, we randomly sample $|V| = 20$ nodes to be in the training set, while the rest of the nodes are in the test set. We repeat each trial for 100 independent runs, implying a different graph structure, node features and GCN weight initialization in each run.

Results. In Fig. 2 (bottom), for each strategy, we report the statistical parity difference after training the GCN across varying perturbation rates $\delta \in [0.05, 0.3]$ with steps of 0.025. We find that (1) DD strategy increases $SPD$, (2) DE strategy fluctuates around 0, (3) ED strategy also fluctuates around 0 without a significant influence on $SPD$, and (4) EE strategy decreases $SPD$ but at a lower rate than DD. To reflect on our qualitative analysis in III-C, we report the changes in error rates from Equation 3 in the supplemental information.

IV. EXPERIMENTS

In this section, we evaluate FA-GNN on three real-world social network datasets. Specifically, we aim to explore:

1) Effectiveness: How do the proposed FA-GNN strategies perform in degrading the fairness of the GNN models?
2) Efficiency: What is the impact of the amount of labeled data available to the attacker on the fairness results?

3) Comparative analysis: How do existing heuristics and accuracy-targeting attack methods influence GNN’s fairness compared to FA-GNN? Does FA-GNN (unintentionally) degrade accuracy in comparison?

Datasets. We utilize three empirical datasets (Table I). 1) Pokec_{z} and 2) Pokec_{n} [4] are two social networks sampled from Pokec [20], a popular social network in Slovakia. 3) DBLP [21] is a coauthor bibliography network in computer science.

Fairness attack. We compare four adversarial linking strategies of FA-GNN, i.e., DD, DE, ED, EE, on all four subsets in $\mathcal{S}$, with the total of 10 FA-GNN attacks after removing duplicates. All FA-GNN variants use a surrogate GCN to predict the labels of unlabeled nodes.

Baselines. To validate whether existing attacks on accuracy can (unintentionally) degrade fairness, and to compare the performance of FA-GNN, we consider the following attack methods: i) a random baseline that randomly adds edges to the network, ii) a baseline accuracy-targeting attack method DICE [23], and iii) a SoTA accuracy-targeting attack method PR-BCD [7] that can scale to large graphs. Note that the space requirements of attacks such as [2], [24], [26], [30] are prohibitive given the sizes of our datasets, so we only consider scalable attacks for our comparison.

Evaluation. After applying the attack to the graph data, we train a victim GCN, which is independent of the surrogate GCN used by FA-GNN. Then we evaluate the effectiveness of attacks using $SPD$, where farther values from 0 signal more unfairness. A more comprehensive evaluation using other group fairness metrics and other victim GNN models can be found in the supplemental information.

Experimental settings. In the experiments, we increase the attack perturbation rate from 0.05 to 0.3 with a 0.05 increment. For the victim GNN model training, we randomly pick 50%, 25%, 25% of labeled nodes as training set, validation set, and testing set. For the attacker, the available labeled nodes are the same as the victim GNN’s training set. We report the

| Dataset | Pokec_{z} | Pokec_{n} | DBLP |
|---------|-----------|-----------|------|
| # of nodes | 67,435 | 66,082 | 20,111 |
| # of edges | 617,765 | 516,784 | 57,508 |
| Feature dimension | 274 | 263 | 2,491 |
| Label ($y$) | working | working | research |
| Sensitive Attribute (s) | region | region | gender |
| % of nodes in $y_0s_0$ | 30.88% | 35.46% | 54.21% |
| % of nodes in $y_0s_1$ | 15.57% | 15.81% | 12.58% |
| % of nodes in $y_1s_0$ | 33.55% | 32.40% | 27.63% |
| % of nodes in $y_1s_1$ | 20.00% | 16.33% | 5.58% |
| Intra-label density* ($\times 10^{-4}$) | 4.7 | 4.8 | 12.5 |
| Intra-group density ($\times 10^{-4}$) | 7.0 | 6.9 | 8.3 |

* The intra density is the number of intra-edges divided by the number of all possible edges.
averaged fairness results on the test set on 5 independent runs. We provide the detailed experimental setting in our repository.

A. Effectiveness of attacks

We evaluate the attack performance on the GCN model based on the signed SPD in Fig. 3, where a perturbation rate of 0 represents the clean graph. With the perturbation rate increasing, the deviation of the curves away from the center line of zero (absolute fairness) indicates increased unfairness. We observe that, on Pokec-z/n, out of the four strategies, DD and EE result in a consistent change in SPD, while DE and ED have no significant influence. On DBLP, EE appears to have a minor effect, while DE causes a clear shift in SPD. Overall, DD is the most effective on all three datasets: DD($y_1 s_1 y_0 s_0$) increases SPD while DD($y_1 s_0 y_0 s_1$) decreases SPD. This finding is particularly interesting since it is plausible that encouraging links among, e.g., different demographic groups, leads to an improvement of some notion of fairness [14], [19]. Similar to the synthetic graphs, the performance of DD strategy drops for high perturbation rates. We also attribute that to the high heterophily rate (or “good heterophily”) attained at this stage.

Our extended evaluation on other fairness metrics (equal opportunity and equalized odds [8]) show the same trends. Our extended evaluation also shows that FA-GNN degrades the fairness of various GNN models, including fairness-enhancing models, such as NIFTY [1] and FairGNN [4]. This shows the effectiveness and robustness of FA-GNN in degrading various GNNs’ fairness on various fairness metrics.

B. Efficiency of attacks

We evaluate the efficiency of FA-GNN in terms of the amount of labeled data available to the attacker. We consider varying percentages of labeled nodes available to the surrogate GCN: from 10% to 50% with a 10% increment. We keep the ratios of labeled data for the victim GCN unchanged: 50% training, 25% validation and 25% testing. We fix the perturbation rate at 0.15. We report the statistical parity of the GCN’s predictions in Fig. 4.

From the results, we observe that, 1) efficiency: with a small percent of available labeled data, i.e., 10%, FA-GNN already degrades the GCN’s fairness to a large extent. This indicates that FA-GNN can efficiently harm GNN’s fairness with very limited access to the information on true labels; 2) influence of labeled data: there is no obvious increase in SPD with increasing number of labeled nodes, except for DD strategy on Pokec-z/n, where the performance slightly improves with more labeled nodes. This demonstrates that the availability of more labeled nodes does not necessarily lead to more successful fairness attacks.

C. Comparative analysis of attacks

We compare the baseline attack methods (Random, DICE and PR-BCD) to FA-GNN’s DD strategy, i.e., $y_1 s_1$-DD and $y_1 s_0$-DD. We report both absolute statistical parity difference $|SPD|$ and node classification accuracy on empirical datasets in Fig. 5. Higher $|SPD|$ implies more unfair predictions.

From the results, we observe that: 1) fairness: Random and accuracy-targeting attack methods have only little impact on the fairness results in terms of $|SPD|$ score compared to FA-GNN strategies; 2) accuracy: While Random, DICE, and FA-GNN show no significant drop in accuracy, the SotA accuracy-targeting PR-BCD significantly degrades model performance with a large drop in accuracy. FA-GNN only has a marginal effect on the accuracy across all the datasets. This indicates that these fairness attacks are less likely to be detected by a defender (more deceptive), at least based on model performance. As a result, if model fairness is not monitored, fairness attacks can go unnoticed, exacerbating the attack consequences for certain groups.

V. CONCLUSION

In this paper, we presented adversarial attack strategies that target the fairness results of graph neural network (GNN) based node classifier models. We started by defining fairness attacks on GNNs and describing adversarial strategies that
Our work demonstrates that fairness attacks significantly degrade the fairness of node classification results. We illustrated these consequences on synthetically generated graph datasets. Our evaluations on empirical datasets showed that fairness attacks significantly degrade the fairness of node classification results without a significant drop in accuracy. Our work demonstrates the vulnerability of GNNs to simple structural perturbation without a significant drop in accuracy. We illustrated these consequences on synthetically generated graph datasets. Our evaluations on empirical datasets showed that fairness attacks significantly degrade the fairness of node classification results without a significant drop in accuracy. Our work demonstrates the vulnerability of GNNs to simple structural perturbation based adversarial attacks on fairness. Designing methods that improve the fairness of GNNs while being robust against such adversarial attacks should be a pressing concern for future research.

REFERENCES

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

[2] J. Chen, Y. Chen, H. Zheng, S. Shen, S. Yu, D. Zhang, and Q. Xuan, “Mga: momentum gradient attack on network,” IEEE Transactions on Computational Social Systems, vol. 8, no. 1, pp. 99–109, 2020.

[3] A. Chhabra, A. Singla, and P. Mohapatra, “Fairness degrading adversarial attacks against clustering algorithms,” arXiv preprint arXiv:2110.12020, 2021.

[4] E. Dai and S. Wang, “Say no to the discrimination: Learning fair graph neural networks with limited sensitive attribute information,” in Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021, pp. 680–688.

[5] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song, “Adversarial attack on graph structured data,” in International conference on machine learning. PMLR, 2018, pp. 1115–1124.

[6] C. Dwork, M. Hardt, T. Pittasi, O. Reingold, and R. Zemel, “Fairness through awareness,” in Proceedings of the 3rd innovations in theoretical computer science conference, 2012, pp. 214–226.

[7] S. Geisler, T. Schmidt, H. Şirin, D. Zügner, A. Bojchevski, and S. Günnemann, “Robustness of graph neural networks at scale,” Advances in Neural Information Processing Systems, vol. 34, 2021.

[8] M. Hardt, E. Price, and N. Srebro, “Equality of opportunity in supervised learning,” Advances in neural information processing systems, vol. 29, pp. 3315–3323, 2016.

[9] P. W. Holland, K. B. Laskey, and S. Leinhardt, “Stochastic blockmodels: First steps,” Social networks, vol. 5, no. 2, pp. 109–137, 1983.

[10] H. Hussain, T. Duricic, E. Lee, D. Helic, and R. Kern, “The interplay between communities and homophily in semi-supervised classification using graph neural networks,” Applied Network Science, vol. 6, no. 1, pp. 1–26, 2021.

[11] W. Jin, Y. Li, H. Xu, Y. Wang, S. Ji, C. Aggarwal, and J. Tang, “Adversarial attacks and defenses on graphs: A review, tool, and empirical studies,” arXiv preprint arXiv:2003.00653, 2020.

[12] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” arXiv preprint arXiv:1609.02907, 2016.

[13] Y. Ma, S. Wang, T. Derr, L. Wu, and J. Tang, “Attacking graph convolutional networks via rewiring,” arXiv preprint arXiv:1906.03750, 2019.

[14] F. Masrour, T. Wilson, H. Yan, P.-N. Tan, and A. Esfahanian, “Bursting the filter bubble: Fairness-aware network link prediction,” in Proceedings of the AAAI conference on artificial intelligence, vol. 34, no. 01, 2020, pp. 841–848.

[15] M. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, “A survey on bias and fairness in machine learning,” ACM Comput. Surv., vol. 54, no. 6, jul 2021. [Online]. Available: https://doi.org/10.1145/3457607

[16] N. Mehrabi, M. Naveed, F. Morstatter, and A. Galstyan, “Exacerbating algorithmic bias through fairness attacks,” arXiv preprint arXiv:2012.08723, 2020.

[17] V. Nanda, S. Dooley, S. Singla, S. Feizi, and J. P. Dickerson, “Fairness through robustness: Investigating robustness disparity in deep learning,” in Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 2021, pp. 466–477.

[18] D. Solans, B. Biggio, and C. Castillo, “Poisoning attacks on algorithmic fairness,” arXiv preprint arXiv:2004.07401, 2020.

[19] I. Spinelli, S. Scardapane, A. Hussain, and A. Uncini, “Fairdrop: Biased edge dropout for enhancing fairness in graph representation learning,” IEEE Transactions on Artificial Intelligence, 2021.

[20] L. Takac and M. Zabovský, “Data analysis in public social networks,” in International scientific conference and international workshop present day trends of innovations, vol. 1, no. 6, 2012.

[21] L. Tang, H. Liu, J. Zhang, and Z. Nazeri, “Community evolution in dynamic multi-mode networks,” in Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 2008, pp. 677–685.

[22] B. Wang and N. Z. Gong, “Attacking graph-based classification via manipulating the graph structure,” in Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019, pp. 2023–2040.

[23] M. Waniek, T. P. Michalak, T. Rahwan, and M. J. Wooldridge, “Hiding individuals and communities in a social network,” CoRR, vol. abs/1608.00375, 2016. [Online]. Available: http://arxiv.org/abs/1608.00375

[24] H. Wu, C. Wang, Y. Tsyshetskii, A. Docherty, K. Lu, and L. Zhu, “Adversarial examples on graph data: Deep insights into attack and defense,” arXiv preprint arXiv:1903.01610, 2019.

[25] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” IEEE transactions on neural networks and learning systems, vol. 32, no. 1, pp. 4–24, 2020.

[26] K. Xu, H. Chen, S. Liu, P.-Y. Chen, T.-W. Weng, M. Hong, and X. Lin, “Topology attack and defense for graph neural networks: An optimization perspective,” in Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, International Joint Conferences on Artificial Intelligence Organization, 7 2019, pp. 3961–3967. [Online]. Available: https://doi.org/10.24963/ijcai.2019/550

[27] X. Zhang and M. Zitnik, “Ginguard: Defending graph neural networks against adversarial attacks,” Advances in Neural Information Processing Systems, vol. 33, 2020.

[28] J. Zhu, Y. Yan, L. Zhao, M. Heimann, L. Akoglu, and D. Koutra, “Beyond homophily in graph neural networks: Current limitations and effective designs,” Advances in neural information processing systems, 2020.

[29] D. Zügner, A. Akbarnejad, and S. Günnemann, “Adversarial attacks on neural networks for graph data,” in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 2847–2856.

[30] D. Zügner and S. Günnemann, “Adversarial attacks on graph neural networks via meta learning,” in International Conference on Learning Representations, 2018.