Mitigating Multisource Biases in Graph Neural Networks via Real Counterfactual Samples

Zichong Wang[1], Giri Narasimhan[1], Xin Yao[2] and Wenbin Zhang[1*]
[1]Florida International University, U.S.A.
[2]Southern University of Science and Technology, China
Email:{zwang114,giri,wenbin.zhang}@fiu.edu, xiny@sustech.edu.cn

2023 IEEE International Conference on Data Mining (ICDM) | 979-8-3503-0788-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDM58522.2023.00073

***Abstract*—Graph neural networks (GNNs) have demonstrated remarkable success in various real-world applications. However, they often inadvertently inherit and amplify existing societal bias. Most existing approaches for fair GNNs tackle this bias issue by assuming that discrimination solely arises from sensitive attributes such as race or gender, while disregarding the prevalent labeling bias that exists in real-world scenarios. Additionally, prior works attempting to address label bias through counterfactual fairness often fail to consider the veracity of counterfactual samples. This paper aims to bridge these gaps by investigating the identification of authentic counterfactual samples within complex graph structures and proposing strategies for mitigating labeling bias guided by causal analysis. Our proposed learning model, known as Real Fair Counterfactual GNNs (RFCGNN), also goes a step further by considering the learning disparity resulting from imbalanced data distribution across different demographic groups in the graph. Extensive experiments conducted on three real-world datasets and a synthetic dataset demonstrate the effectiveness and practicality of the proposed RFCGNN approach.**

***Index Terms*—Counterfactual fairness, graph learning, real counterfactual samples**

I. Introduction

Graph data is ubiquitously present across numerous real-world application scenarios, including financial markets [1], biological networks [2], and social networks [3]. This wide availability and diversity of graph-based data in practical applications have fueled the remarkable advancements in graph representation learning. Consequently, it has inspired the development of numerous graph learning paradigms in recent years. Among these, Graph Neural Networks (GNNs) stand out as a foundational and extensively utilized approach, consistently delivering exceptional performance across a diverse range of tasks and applications [4]. Generally, GNNs employ a message-passing mechanism that updates a node's representation through iterative aggregation of its neighboring nodes' representations. This process retains both nodal features and the graph structure information, effectively supporting a variety of downstream tasks. Although GNNs demonstrate strong performance, they can inadvertently inherit and even exacerbate biases embedded in the training data, leading to algorithmic bias against certain deprived subgroups in the population [5]. Such biased predictions give rise to ethical and societal concerns, significantly constraining the use of GNNs in high-stakes decision-making systems, such as job applicant ranking [6], healthcare [7] and credit scoring [8]. For instance, if a bank's decision on a loan application was swayed by the applicant's race and that of their close contacts, it would constitute a serious ethical problem [9].

To promote fairness in GNNs, a majority of existing approaches utilize notions of statistical fairness to evaluate and address any unfairness in node representation learning on graphs [10]–[12]. The motivation behind these methods is to achieve fair GNNs by achieving statistically fair predictions for different subgroups or individuals, with these methods primarily focusing on *sensitive attributes* such as race or gender as the only source of bias [13]. However, these approaches fall short when it comes to quantifying and mitigating discrimination in scenarios where labeling bias is present [14]. Labeling bias occurs when societal biases, prejudices, or discriminatory practices influence the data collection process, causing labels or outcomes to depict more than the objective phenomena they are intended to measure [15]. This distorted representation of reality introduces systemic biases into the training data, which in turn can be learned and perpetuated by GNNs, making them less biased towards deprived subgroups as identified by the sensitive attribute. To tackle this issue, recent studies have incorporated the concept of counterfactual fairness into graph-structured data. Different from the statistical notion of fairness, instead of merely statistically balancing predictions, counterfactual fairness seeks to address the root causes of inequity by modeling the underlying causal structure of variables. Its goal is to ensure that different versions (*i.e.,* counterfactual examples) of the similar individual receive similar predictions. For instance, in a job recruitment process, two candidates with similar qualifications and experience should have a similar likelihood of being hired, regardless of their gender.

Existing work on graph counterfactual fairness typically generates counterfactual samples either by directly flipping sensitive attributes or through the employment of graph generation models. For instance, NIFTY [16] creates counterfactuals by introducing perturbations to sensitive attributes, and it seeks to maximize the likeness between original and perturbed representations to ensure invariance to these sensitive attributes. On the other hand, GEAR [9], utilizes GraphVAE to create counterfactuals. It aims to minimize the discrepancy between original and counterfactual representations to eliminate the impact of sensitive attributes. A significant limitation of these approaches is their failure to consider the generation of

*Corresponding author

979-8-3503-0788-7/23/$31.00 ©2023 IEEE
DOI 10.1109/ICDM58522.2023.00073

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

counterfactual samples that may not be feasible or realistic. Specifically, merely altering sensitive attributes fails to account for the causal influence these attributes may have on other features or the graph structure, thereby not truly embodying counterfactual examples. Moreover, generative methods do not provide sufficient checks on the realism of the counterfactuals and could be over-complicated.

In this paper, we explore the domain of graph-based counterfactual fairness, paying attention to potential causal relationships between each sample and its neighboring nodes. In particular, we examine the changes arising from potential causal effects on the nodes and their neighbors due to variations in the samples' sensitive attributes. Additionally, we examine the feasibility of deriving counterfactual samples directly within the given graph instead. This remains a largely unexplored field, teeming with distinctive challenges: **i) Complexity of Counterfactual Graph Data Structures:** Unlike tabular data, graph data do not follow the principle of Independent and Identically Distributed (IID), due to the inherently interconnected relationships between each node and its neighbors. Thus, given the complexity of these relationships, counterfactual samples necessitate changes not only to the nodes themselves but also to their connectivity with neighboring nodes [17]. **ii) Realistic Counterfactual Discovery:** Existing work on graph counterfactual fairness generates counterfactual samples by directly flipping sensitive attributes or perturbing them. This unsupervised approach, however, neglects the possibility of generating counterfactual samples that might be unrealistic [18]. **iii) Learning the sources of bias for bias-related representation.** Due to the challenges posed by correlations among attributes in distinguishing bias-relevant and bias-irrelevant representations, existing fairness methods apply fairness constraints to the entire representation space, encompassing both sensitive and non-sensitive attributes [19]. However, this approach can lead to unnecessary information loss and adversarial bias. Consequently, there is a strong benefit in learning bias-related representations to mitigate these issues effectively which requires sophisticated design.

To tackle the above challenges, we introduce a unique framework that leverages the concept of counterfactual thinking to ensure fairness in graph-based, socially sensitive decision-making contexts. *To the best of our knowledge, this is the first work that utilizes authentic counterfactual samples to simultaneously alleviate multisource biases arising from sensitive attributes and the labeling process in graph-based models*. Specifically, we conduct a causal analysis on the original samples and counterfactual samples, deriving several constraints to ensure the learned representations maintain invariance across a range of sensitive attributes while preserving as much node information as possible. To tackle the issue of realistic counterfactual discovery, we incorporate both labels and sensitive attributes in our methodology, filtering potential counterfactuals to guarantee their authenticity and relevance. Furthermore, we introduce an innovative approach for learning the sources of bias for bias-related representations. Instead of applying fairness constraints universally, we discern between bias-relevant and bias-irrelevant representations, focusing our fairness enforcement efforts where they are truly necessary. This approach prevents unnecessary information loss and adversarial bias, leading to a more balanced and effective model. The key contributions of this paper are therefore as follows:

- We introduce a novel causal formulation that paves the way for understanding the generation process of graph structures and the fair learning task of node representation.
- We present a novel causal framework, termed *Real Fair Counterfactual GNNs (RFCGNN)*, which utilizes authentic counterfactual samples to alleviate multisource biases arising from sensitive attributes and labeling processes in graph-based models. Specifically, RFCGNN performs a causal analysis on both original and counterfactual samples, generating constraints that ensure invariance across different sensitive attributes while maximizing the preservation of node information. This approach enables our model to enhance fairness without compromising performance.
- We conduct comprehensive testing on three real-world datasets and one synthetic dataset to demonstrate the efficacy of our model in balancing fairness and prediction accuracy.

The remainder of this paper is structured as follows: Section II provides an overview of relevant literature. Notations is presented in Section III. Our proposed causal model and RFCGNN are detailed in Section IV. Section V describes the experimental setup and offers an analysis of the results. Lastly, we conclude the paper in Section VI.

II. Related work

A. Graph Neural Networks

Graph neural networks (GNNs) have found widespread utility in various tasks involving graph-structured data. These tasks include, but are not limited to, node classification [11], [20], [21], graph classification [22], [23], and link prediction [12], [24]. The notable performance of GNNs across these tasks has significantly expanded their application domains. For instance, healthcare organizations can leverage GNNs to analyze patient networks to inform critical decisions like organ transplants [25]. However, the deployment of GNNs in such high-stakes decision-making scenarios necessitates additional considerations such as fairness [26]. For instance, in the healthcare context, decisions informed by GNNs need to be not only accurate but also fair, given the potential life or death implications. Given these requirements, there is an emerging trend in the research community to develop GNN models that incorporate fairness considerations for handling graph-based tasks [27], [28].

B. Fairness on graphs

Fairness in machine learning, particularly in graph-based models, has become a crucial area of research in recent years [5], [29]–[32]. Researchers have explored numerous

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

fairness notions, including group fairness [33]–[35], individual fairness [36]–[38], and counterfactual fairness [18], [39], [40]. These metrics can be applied to enforce and evaluate the fairness performance of Graph Neural Networks (GNNs). Recent studies have proposed methods to enhance fairness in GNN learning [9], [16], [41]. Most of them are based on common fairness metrics, such as statistical parity or equal opportunity, and aim to enforce statistical equity between sensitive attributes and predictions generated from the learned representations. However, these methods often fall short in addressing labeling bias as they tend to overlook biases in graph structures or labels that could stem from the causal influence of the sensitive attribute. Some research has attempted to extend counterfactual fairness to graph learning, but their use of encoder-decoder frameworks can lead to increased information loss and subsequent performance degradation.

Contrary to previous research, our approach seeks to mitigate these issues. Specifically, we focus on counterfactual fairness in graph learning and propose a unique approach to select real counterfactual samples with similar characteristics as guiding principles, thereby avoiding the generation of synthetic counterfactual samples. This allows us to mitigate biases in graph-based models more effectively without compromising model performance.

III. Notation

We formalize the fair graph neural networks problem in the context of undirected and unweighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X})$, where the set of nodes is denoted as $\mathcal{V}$, the set of edges is denoted as $\mathcal{E}$ ($\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$), and the set of node features is denoted as $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ (n = $|V|$), where $x_i \in \mathbb{R}^{1\times D}$ represents the features of each node i. We let A denote the adjacency matrix of the graph $\mathcal{G}$, where $A_{i,j}$ takes on the value 1 if there exists an edge $i \rightarrow j$, and 0 otherwise. Meanwhile, each node v_i has a sensitive attribute, we utilize S $\in \{0,1\}^{N\times 1}$ to represent the sensitive attributes, where s_i represents whether or not a given individual v_i is a member of the deprived set. Note that $s_i \in \mathcal{X}$. Further, we let G_i represent the ego graph for each node v_i. The ego graph provides a focused view of the direct neighbors and interconnections around a specific node within a larger network. Without loss of generality, we let $\mathcal{L} = \{v_1, v_2, \ldots, v_L\}$ denote the set of $\mathcal{L}$ labeled vertices, with associated ground-truth labels Y = $\{y_1, \ldots, y_L\}$, where y_i represents the ground-truth label for the vertex v_i. Moreover, $\mathcal{U} = \{v_{L+1}, v_{L+2}, \ldots, v_{L+U}\}$ denotes the set of $\mathcal{U}$ unlabeled vertices, and the prediction label for an unlabeled vertex is denoted as $\hat{y}$. Please note $\mathcal{L} \bigcup \mathcal{U} = \mathcal{V}$. In addition, in this study we assume both sensitive attributes and labels are binary variables for convenience.

IV. Methodology

In this section, we begin by presenting our causal model, followed by an overview of the RFCGNN workflow. Subsequently, we provide an in-depth explanation of the essential elements comprising the model. Finally, we introduction the objective function utilized for learning.

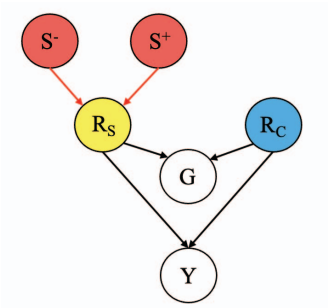

Fig. 1: The structural causal model of RFCGNN.

A. Causal Model

We first introduce our causal model, which forms the crux of our approach to addressing bias in graph learning. Our study is primarily concerned with the case of multisource biases, a scenario that brings to light inherent flaws in fairness concepts solely based on statistics. As a consequence, we move beyond statistical measures and adopt a causality-based fairness concept. This shift necessitates a detailed causal analysis of the observed graphs, laying the groundwork for our RFCGNN workflow and the objective function for the learning, which we elaborate on in the subsequent sections. Without loss of generality, in this study, we focus on the node classification task and construct a structural causal model as depicted in Figure 1. The crux of our model involves delineating the causal relationships between five sets of elements: different sensitive values (S^- and S^+), ground truth labels (Y), sensitive relational representation (R_S), content representation (R_C), and the ego-graph of each node (G). Specifically, S^- represents sensitive attributes associated with deprived groups that have traditionally been underprivileged (*e.g.,* female), while S^+ denotes sensitive attributes related to favored groups that are typically privileged (*e.g.,* male). In addition, the node representation space is disentangled into two distinct components: content representation (R_C) and sensitive relational representation (R_S) which are sensitive attributes unrelated and related representation, respectively. Each link in the model denotes a deterministic causal relationship between two elements. Their interpretations and design methodology are detailed as below,

- $S^+ \rightarrow R_S \leftarrow S^-$. Distinct sensitive values should be associated with comparable distributions in the representation of R_S. This ensures that essential information from sensitive values is preserved while minimizing inherent biases associated with them. The presence of this substructure embodies this interaction and is designed to facilitate a fair learning process.
- $R_S \perp R_C$. The purpose of this design is to ensure the independence between R_S and R_C through disentanglement, which serves to support subsequent substructures.
- $R_S \rightarrow Y \leftarrow R_C$. While it is important to enforce fairness constraints on R_S, both R_S and R_C should play a role in predicting the target variable Y. To achieve this, a disentangled representation is firstly required to encode

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

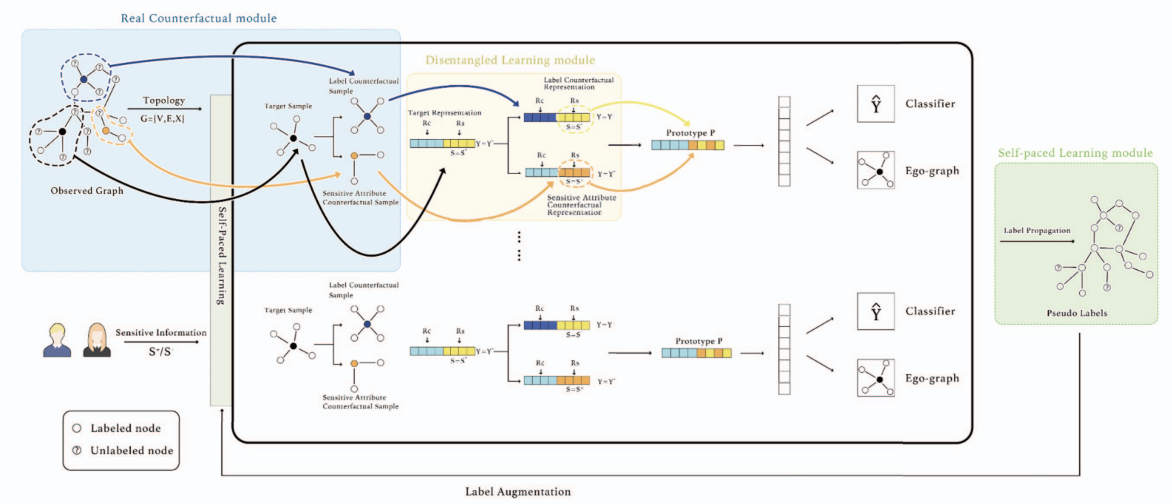

Fig. 2: Overview of the proposed RFCGNN framework.

the original feature space into distinct dimensions for R_S and R_C. This allows for the application of fairness constraints specifically on R_S within this substructure, while preserving as much information about Y as possible.

- $R_S \rightarrow G \leftarrow R_C$. Same as the above substructure but from a graph structure perspective, both R_S and R_C have direct causal effects on G. This substructure facilitates such an accurate and unbiased graph representation learning.

B. RFCGNN: In a Nutshell

Our novel RFCGNN method, inspired by the above causal analysis, is designed to enhance GNN fairness through a unique strategy of using counterfactual samples extracted from observed samples. As Figure 2 shows, RFCGNN comprises three key modules: the Real Counterfactual module (Figure 2 Blue) that finds real counterfactual samples, the Disentangled Learning module (Figure 2 Yellow) capitalizes on these counterfactual samples to segregate the node representations into R_c and R_s. The Self-paced Learning module (Figure 2 Green) which facilitates gradual learning from simpler to more complex instances. Each of these components will be elaborated upon in the subsequent discussion.

C. Real Counterfactual Module

Taking inspiration from causal analysis, the representation of a node v_i is decomposed into R_C and R_S. To ensure an accurate disentanglement of these representations, we utilize counterfactual examples as guiding factors. As illustrated in Figure 2 yellow, within the disentangled learning module, we require two types of counterfactual samples for a given node v_i, observed with a specific sensitive attribute value s_i and a corresponding label y_i: i) the first sensitive attribute counterfactual sample only differs in sensitive attribute ($\mathcal{G}i^S$), which is reflected by similar R_C and identical Y but dissimilar R_S (refer to the yellow and orange bars in Figure 2's yellow section). ii) the second label counterfactual sample only differs in class label $\mathcal{G}_i^L$, which is reflected by similar R_S but dissimilar R_C and Y (refer to the light and dark blue bars in Figure 2's yellow section). The task of finding such counterfactual samples can be formalized as shown in Equation 1:

$$\begin{cases} \mathcal{G}_i^L = \arg\min_{\mathcal{G}_j \in \mathbb{G}} \{d(\mathcal{G}_i, \mathcal{G}_j) | y_i \neq y_j, s_i = s_j\} \\ \mathcal{G}_i^S = \arg\min_{\mathcal{G}_j \in \mathbb{G}} \{d(\mathcal{G}_i, \mathcal{G}_j) | y_i = y_j, s_i \neq s_j\} \end{cases} \tag{1}$$

where $\mathbb{G} = \{\mathcal{G}_i | , v_i \in \mathcal{V}\}$ and $d(\cdot)$ measures the distance between pairs of ego-graphs.

Existing methods for generating graph counterfactual samples often rely on directly flipping sensitive attributes or perturbing features, which may ignore the authenticity of the generated counterfactual samples. To end this, we propose to identify real counterfactual candidates from observed factual graphs instead. This is inspired by hypothetical situations like a female job applicant pondering, 'If I were male, would my application have been rejected? What sets me apart from the accepted applicant?' This line of questioning leads us to identify counterfactuals directly from observed samples, rather than through perturbation or generation. This approach offers two key advantages: it avoids making assumptions about how graphs that include sensitive attributes are generated and eliminates the need for additional supervised signals to select counterfactuals.

However, the direct search for real counterfactual samples presents a challenge: efficiently selecting appropriate counterfactual samples from the interconnected graph data. Considering the complexity of graph structures and the vast search space of graph data, computing pairwise distances between ego-graphs becomes highly inefficient and impractical. To address this issue, we propose measuring distances in the latent space, leveraging the captured graph structure and node attribute information to enhance computational efficiency. The task in Equation 1 in thus reformulated as:

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

$$\begin{cases} C_i^L = \arg\min_{z_j \in Z} \{\|z_i - z_j\|_2^2 | y_i \neq y_j, s_i = s_j\} \\ C_i^S = \arg\min_{z_j \in Z} \{\|z_i - z_j\|_2^2 | y_i = y_j, s_i \neq s_j\} \end{cases} \tag{2}$$

where $Z = \{z_i|, v_i \in \mathcal{V}\}$ is learned representation matrix and the L2 distance is employed. Note that for each factual input, two sets of counterfactual samples are obtained instead of two samples. Consequently, the counterfactual C^L can naturally extend to a set of label counterfactual samples consisting of k instances $\{C_i^L | i = 1, \ldots, k\}$, and C^S can be extended as a set of sensitive attribute counterfactual samples also comprising k instances $\{C_i^S | j = 1, \ldots, k\}$, where k is a constant number.

D. Disentangled Learning module

As described in the causal analysis model in Section IV-A, we need to disentangle R_C and R_S in the representation space. Both of these representations contain information useful graph information. However, while R_C should be independent of sensitive attributes, R_S should maintain a strong relevance to them. By distinguishing these two sub-representations, R_C and R_S, we aim to eliminate the impact of sensitive attribute information. We achieve this by ensuring different sensitive attribute values map to R_S with similar probability distributions, thus promoting fair predictions.

To effectively disentangle R_C from R_S, the identified counterfactual samples, according to the methodology in Section IV-C, are leveraged to guide the decomposition of the representations. Initially, the encoder $\Phi(\cdot)$, which maps each node v_i to a latent representation, is trained. The learned representations for the n nodes are denoted by $Z = \{z_1, z_2, \ldots, z_n\}$, where each $z_i \in \mathbb{R}^{1\times d}$ represents the node v_i, and d is the dimensionality of the node representations. As shown in Equation 3, this process transforms information in the input feature matrix X and adjacency matrix A, both of dimensions $n \times d$ and $n \times n$, respectively, into the node representation matrix Z of dimensions $n \times d$.

$$Z = \Phi\ (X, A) = \mathbb{R}^{n\times d} \times \mathbb{R}^{n\times n} = \mathbb{R}^{n\times d} \tag{3}$$

To facilitate the disentanglement of representations during the process, the following four specific constraints are defined:

i) Orthogonality Constraint $\mathcal{J}_D$. This constraint aims to ensure the orthogonality between the representations R_C and R_S, meaning the information contained in R_C should not leak into R_S and vice versa. As previously discussed, R_C is intended to be sensitive attributes irrelevant, while R_S is directly related to them. Thus, the orthogonality between these two representations would contribute to the fairness of the model. Mathematically, this can be expressed as the dot product between vectors from the two representation spaces equal to zero, *i.e.,* $R_C^T \cdot R_S = 0$. However, attaining this level of disentanglement presents a significant challenge due to potential correlations that may exist between sensitive and non-sensitive attributes. To address this challenge, we use the counterfactual examples as a guide. As illustrated in Figure 3, for a sensitive attribute counterfactual sample, its content representation R_C is similar to the target sample, while its sensitive representation R_S^S differs from that of the target sample R_S. Conversely, for a label counterfactual sample, the sensitive representation R_S mirrors that of the target sample, while the content representation R_C^L is distinct. This method effectively avoids the failure of representation separation that could arise if guided solely by a single counterfactual example. Consider a scenario where only label counterfactual sample is used to guide the disentanglement of representation learning. In such a case, both content and sensitive information could exist within the content representation R_C and the counterfactual content representation R_C^L. This overlap could lead to a failure in achieving a proper disentanglement. In order to encourage the representation spaces of R_C and R_S to store distinct information, the following Orthogonal Constraint is defined:

$$\mathcal{J}_D = \arg\min \frac{1}{|\mathcal{V}| \times K} \sum_{v_i \in \mathcal{V}} \sum_{k=1}^{K} \Big[d(R_{C_i}, R_{C_i}^L) + d(R_{S_i}, R_{S_i}^S) + \rho K \times |\cos(R_{C_i}, R_{S_i})| \Big] \tag{4}$$

where $d(\cdot)$ is a distance metric, and $|\cos(R_{C_i}, R_{S_i})|$ is the absolute value of the cosine similarity between vectors R_C and R_S, which we aim to optimize such that it approximates to zero. ρ is the hyperparameter that regulates the degree of the orthogonal constraint. Here, R_{C_i} and R_{S_i} can represent the projections of the i^{th} instance onto the R_C and R_S spaces respectively. This equation is implemented to minimize the cosine similarity between the vectors, thereby maintaining orthogonality and ensuring fairness in the model.

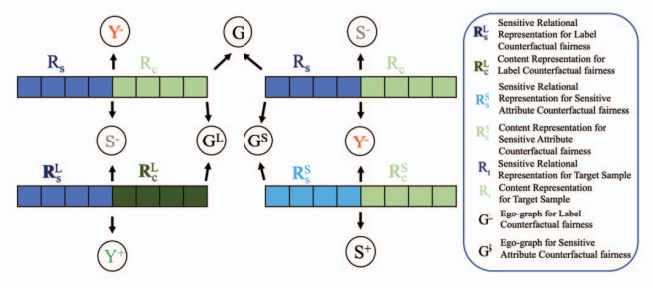

Fig. 3: Real counterfactual samples illustrated.

ii) Fairness Constraint $\mathcal{J}_F$. The goal of this constraint is to desensitize the sensitive attribute-related representation R_S. To this end, R_S is mapped to a new representation space. This new space does not retain any information that can identify whether a node belongs to a certain group. Specifically, this new representation is formulated as a probabilistic mapping to a prototype p, where p is a vector of the same length as R_S. We then replace R_S with p as input to the classifier, which is used for predicting the class label of the instance. Essentially, if a prototype p has an equal probability of appearing the deprived (S^-) and favored (S^+) groups, it becomes indeterminable to which group p belongs. Mathematically, this can be expressed

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

as $P(W = w_i|R_S \in S^+) = P(W = w_i|R_S \in S^-)$, where W is a multinomial random variable, and each w_i denote an instance of the set of prototypes.

We induce a natural probability mapping from R_S to W using a Softmax function as follows:

$$P(W = w_i|R_S) = \frac{\exp(-d(R_S, p_{w_i}))}{\sum_{u=1}^{W} \exp(-d(R_S, p_u))} \tag{5}$$

where $d(\cdot)$ is a distance metric (Euclidean distance). Hence, we formally define statistical parity as follows:

$$M_u^+ = \frac{1}{n^+} \sum_{v_i \in S^+} P(W = w|R_S \in S^+) \tag{6}$$

Finally, we define the final loss for J_P as follows:

$$\mathcal{J}_F = \arg\min \sum_{u=1}^{W} |M_u^+ - M_u^-| + \sum_{i=1}^{n} (R_{Si} - \overline{R_{Si}})^2 \tag{7}$$

where $\overline{R_S}$ is the reconstructions of R_S from W. This constraint encourages the model to encode all information contained within the input attributes except for any information that could lead to biased learning.

iii) Informativeness Constraint $\mathcal{J}_P$. For each node v_i, we learn that the representations R_C and R_S should capture important node attributes and neighborhood information, thereby retaining utility for downstream tasks. Hence, for node v_i, we should be able to get accurate label prediction from R_C and R_S (*i.e.,* (R_C, p) $\rightarrow$ y). Note R_S has been replaced with a prototype representation p to promote fairness, as discussed in the above constraint. The objective thus is to minimize the loss of the prediction model, as depicted in Equation 8.

$$\mathcal{J}_P = \arg\min f_{(\theta)} \frac{1}{|\mathcal{V}_L|} \sum_{v_i \in \mathcal{V}_L} -(y_i \log(\hat{y_i}) + (1-y_i) \log(1-\hat{y_i})) \tag{8}$$

where $f_{(\theta)}$ is the classifier to take R_C and p as input and predict the class distribution of v_i, and y_i is the one-hot encoding of ground-truth label of v_i.

iv) Construction Constraint $\mathcal{J}_R$. For each node v_i, we aim to ensure the accurate representation of the node through R_C and R_S by enabling them to reconstruct the original graph $\mathcal{G}_i$. We formalize construction constraints as the reconstruction of the graph structure. Specifically, for every pair of distinct nodes $\{\forall\ v_i, v_j \in \mathcal{V} \mid v_i \neq v_j\}$, we predict the probability of link existence as $p_{ij} = \sigma(z_i z_j^T)$, where $z_i = [R_C, R_S]$ represents the node representation of node v_i. Lastly, we formally define $\mathcal{J}_R$ as follows:

$$\mathcal{J}_R = \arg\min \frac{1}{|\mathcal{E}^+| + |\mathcal{E}^-|} \sum_{e_{ij} \in \mathcal{E}} L(e_{ij}, \hat{e}_{ij}) \tag{9}$$

where $\mathcal{E}^-$ represents the set of sampled negative edges, and $e_{ij} = 1$ if nodes v_i and v_j are connected, and 0 otherwise.

The issue of inadequate supervision in existing graph counterfactual fairness methods can be effectively mitigated by implementing sufficiency constraints. This measure helps prevent the incorporation of false information into the representation, which could otherwise compromise the ability to reconstruct the observed graph $\mathcal{G}_i$, contravening the principles of the Structural Causal Model.

E. Self-paced Learning Module

Label sparsity presents another significant challenge in graph learning, given the limited availability of labels in the training set. Further complicating matters, the deprived groups are typically less represented compared to the favored groups, making the acquisition of label information from deprived groups significantly more costly in practice. To tackle this issue, a self-paced learning module is incorporated to guide the learning process. Drawing on the cognitive principle of progressing from 'easy' to 'hard' concepts, this module computes self-paced vectors h^y at each learning cycle, where $h^y \in \{0,1\}^{n \times 1}$ denotes the self-paced vectors regarding the class y = $\{0,1\}$. These vectors assign pseudo labels to a set of unlabeled vertices, guided by a self-paced threshold and the predictive model obtained from the previous cycle. Subsequently, the model updates the predictive model by learning from the augmented training data (*i.e.,* pseudo labeled data in addition to labeled data) preserved in the updated self-paced vectors. As learning progresses, the self-paced threshold is increased to heighten the learning difficulty, thus enabling the next cycle's self-paced vectors to be updated accordingly.

Intuitively, the self-paced learning threshold λ effectively selects nodes to be labeled. For instance, when $h_i^0 = 1$, it indicates model classifies x_i to class 0 with a high confidence $\log P_{\text{pre}}(\hat{y}_i = 0|x_i) > -\lambda$. In other hand, when $h_i^0 = 0$, it indicates the prediction loss $-\log P_{\text{pre}}(\hat{y}_i = 0|x_i)$ is higher than learning threshold λ. Hence, the closed-form solution of updating h_i^y is as follows:

$$h_i^y = \begin{cases} 1 & \text{if } \log P_{\text{pre}}(\hat{y}_i = \{0,1\}|x_i) < \lambda, \\ 0 & \text{Otherwise.} \end{cases} \tag{10}$$

Overall, when the model classifies a node with high confidence, it signifies a small prediction loss, deeming the node an 'easy' concept. Conversely, if the prediction loss is substantial, the node is classified as a 'hard' concept, and its introduction into the learning process is deferred. This method permits regulation of the learning pace and complexity by incrementally increasing the value of the self-paced learning threshold, λ. Consequently, $\mathcal{J}_L$ and $\mathcal{J}_S$ are defined as follows:

$$\begin{aligned} \mathcal{J}_L + \mathcal{J}_S = \arg\min \sum_{i=1}^{L+U} [h_i^0 \log P_{\text{pre}}(\hat{y}_i = 0|x_i) \\ + h_i^1 \log P_{\text{pre}}(\hat{y}_i = 1|x_i)] - \lambda \sum_{i=1}^{L+U} (h_i^0 + h_i^1) \end{aligned} \tag{11}$$

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

$$
\begin{aligned}
\arg\min \mathcal{J} &= \mathcal{J}_P + \mathcal{J}_R + \mathcal{J}_D + \mathcal{J}_F + \mathcal{J}_L + \mathcal{J}_S \\
&= \underbrace{\frac{1}{|\mathcal{V}_L|}\sum_{v_i\in\mathcal{V}_L} -(y_i\log(\hat{y_i}) + (1-y_i)\log(1-\hat{y_i}))}_{\mathcal{J}_P\text{: prediction model}} + \alpha \underbrace{\frac{1}{|\mathcal{E}^+| + |\mathcal{E}^-|}\sum_{e_{ij}\in\mathcal{E}} L(e_{ij}, \hat{e}_{ij})}_{\mathcal{J}_R\text{: reconstruction model}} \\
&+ \beta \underbrace{\frac{1}{|\mathcal{V}| \times K}\sum_{v_i\in\mathcal{V}}\sum_{k=1}^{K}[d(R_C, R_C^L) + d(R_S, R_S^S) + \rho K \times |\cos(R_C, R_S)|]}_{\mathcal{J}_D\text{: disentangled model}} + \omega \underbrace{\sum_{u=1}^{W}|M_u^+ - M_u^-| + \sum_{i=1}^{n}(R_{Si} - \overline{R_{Si}})^2}_{\mathcal{J}_F\text{: fair representations model}} \\
&+ \gamma \underbrace{\sum_{i=1}^{L+U}[h_i^0 \log P_{\text{pre}}(\hat{y}_i = 0|x_i) + h_i^1 \log P_{\text{pre}}(\hat{y}_i = 1|x_i)]}_{\mathcal{J}_L\text{: label propagation model}} - \lambda \underbrace{\sum_{i=1}^{L+U}(h_i^0 + h_i^1)}_{\mathcal{J}_S\text{: self-paced learning}}
\end{aligned} \tag{12}
$$

F. Final Generic Joint Learning Framework

The final objective function of the proposed RFCGNN framework, as presented in Equation 12, brings together the aforementioned three modules. This function consists of six parts and is governed by the hyperparameters α, β, ω, γ, and λ, which are responsible for balancing the contributions of the various elements in the overall objective function. The first term, $\mathcal{J}_P$, aims to minimize the prediction loss. The next term, $\mathcal{J}_R$, works to minimize the reconstruction loss for the representations R_C and R_S, with the function $L(\cdot)$ denoting cross-entropy. Following this, $\mathcal{J}_D$ encourages the learned representations to distinguish between sensitive relevant and irrelevant representation. The fourth component, $\mathcal{J}_F$, is the fairness module, aiming to desensitize the representation R_S, thereby promoting fairness in model predictions. The fifth term, $\mathcal{J}_L$, corresponds to the label propagation model that seeks to maximize the likelihood of observing x_i in its predicted class $\hat{y}_i$. The final term represents the self-paced regularizer, which aids in managing the learning pace of label propagation at a global level.

V. Experiment

A. Datasets

We conduct experiments on three real-world datasets and one synthetic dataset: i) The **German** dataset [42] contains the credit information of clients in a German bank. In this graph, each node represents a client, and each edge between a pair of nodes illustrates the similarity of their credit accounts. The sensitive attribute is the clients' gender, with the aim of classifying clients into good versus bad credit risks. ii) The **Credit** dataset [43] contains individuals' default payment information. In this graph, every node signifies an individual, while every edge between a pair of nodes indicates the similarity in their expenditure and payment patterns. The sensitive attribute is age and the objective of predicting whether their default mode of payment is a credit card. iii) The **Bail** dataset [16] presents data related to defendants who were granted bail in U.S. state courts. In this context, each node corresponds to a defendant, while an edge connecting two nodes signifies similarities in their criminal records and demographic details. The race of the defendants is employed as the sensitive attribute in our analysis. The ultimate goal is to classify defendants into two categories: those suitable for bail and those who should not be bailed. iv) The **Synthetic** dataset, we create a synthetic dataset using a causal model as shown in Figure 1. In our setup, we have binary sensitive attributes and labels, which are generated based on a Bernoulli distribution. Our model allows us to manipulate various parameters including the sensitive attribute probability, label probability, and feature dimensions. For each node, we combine content and latent sensitive relational features to create a latent feature, then use these latent features to generate the observable features. Furthermore, the existence of edges between nodes is determined by potential feature similarities between two nodes, taken in conjunction with Gaussian noise. Table I details the characteristics of them.

TABLE I: Summary of the datasets used in the evaluations.

Dataset	German	Credit	Bail	Synthetic
Vertices	1,000	30,000	18,876	2,000
Edges	21,742	137,377	311,870	4,570
Feature dimension	27	13	18	25
Average Degree	44.5	10	34	4.9
Sensitive Attribute	Gender	Age	Race	Gender

B. Baselines

To benchmark the performance of our method, we compare against seven state-of-the-art methods: Graph Convolutional Networks (GCN) [11], GIN [44], GraphSAGE [45], FairGNN [28], EDITS [46], NIFTY [16], and GEAR [9]. The first three are plain node classification methods. These methods primarily focus on the node classification task and do not incorporate explicit designs for fairness constraints. The next two are fair node classification methods, specifically designed to make accurate predictions while adhering to group fairness constraints. The final two are graph counterfactual fairness methods, specifically engineered to achieve graph counterfactual fairness in addition to making accurate node

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

classification predictions. By comparing our proposed method, RFCGNN, with these different types of baselines, we can thoroughly assess its performance from various aspects.

C. Evaluation Metrics

The evaluation involves two fairness metrics and two ML performance metrics. We will first present the fairness measures, then the ML performance metrics.

a) Fairness metrics: To evaluate our model fairness, we employed two commonly used [9] fairness metrics: Statistical Parity Difference (SPD) and Equal Opportunity Difference (EOD). The former quantifies the disparity in probabilities of receiving a benefit between the favored group and the deprived group, and is given by $SPD = P\left[\hat{Y} = 1|S = 0\right] - P\left[\hat{Y} = 1|S = 1\right]$, while the latter measures the difference in True Positive Rates (TPR) between the favored and deprived groups, and is calculated as $EOD = P\left[\hat{Y} = 1|S = 0, Y = 1\right] - P\left[\hat{Y} = 1|S = 1, Y = 1\right]$. Note that the absolute value of all these metrics are used, in which a value of zero represents optimal fairness and higher values indicate a greater level of bias.

b) Performance metrics: To evaluate model performance, two performance metrics, accuracy, and F1-Score, are employed. The higher the values of these metrics, the better the performance.

D. Experiment Results

The results are structured around four research questions.

RQ1: What is the effectiveness of the proposed RFCGNN framework when applied to fair node classification tasks on real-world datasets?

This Research question evaluates the effectiveness of RFCGNN and existing methods in three real-world datasets. The responses to RQ1 are informed by the results shown in Table II. The empirical results presented in Table II strongly affirm the effectiveness of our proposed RFCGNN framework. For all three real-world datasets, the proposed RFCGNN framework demonstrates competitive or superior performance compared to other methods, both in terms of node classification performance and group fairness metrics. Specifically, in terms of group fairness, compared to other baselines, RFCGNN consistently ranks top across all datasets. Furthermore, in terms of performance, RFCGNN ranks three times in first and three times in second place (indicated by the darker and lighter blue cells respectively) across all three datasets. This indicates that the proposed framework can achieve high prediction performance, which is critical for any node classification task. In summary, our proposed method not only sustains a performance level akin to that of standard node classification methods, but it also mitigates bias more effectively than existing fair node classification methods, thus establishing it as a balanced and robust choice for real-world socially sensitive applications.

RQ2: Is the proposed RFCGNN capable of identifying suitable counterfactuals?

TABLE II: Predictive and fairness performance for RFCGNN and baselines across real-world datasets (the darker cells show the top rank and the lighter cells show the second rank).

Dataset	Methods	Accuracy	F1-Score	SPD	EOD
German	GCN	0.73	0.79	0.38	0.31
	GraphSAGE	0.72	0.82	0.26	0.21
	GIN	0.72	0.81	0.18	0.16
	FairGNN	0.69	0.81	0.13	0.08
	EDITS	0.68	0.79	0.14	0.07
	NIFTY	0.71	0.79	0.13	0.07
	GEAR	0.64	0.78	0.08	0.06
	RFCGNN	0.72	0.82	0.06	0.04
Credit	GCN	0.68	0.79	0.12	0.11
	GraphSAGE	0.75	0.81	0.11	0.12
	GIN	0.69	0.79	0.15	0.14
	FairGNN	0.65	0.73	0.18	0.16
	EDITS	0.75	0.82	0.15	0.09
	NIFTY	0.72	0.81	0.11	0.09
	GEAR	0.73	0.81	0.10	0.08
	RFCGNN	0.74	0.85	0.07	0.06
Bail	GCN	0.86	0.74	0.09	0.03
	GraphSAGE	0.89	0.77	0.14	0.11
	GIN	0.83	0.68	0.09	0.08
	FairGNN	0.91	0.78	0.07	0.05
	EDITS	0.85	0.76	0.06	0.07
	NIFTY	0.92	0.75	0.06	0.04
	GEAR	0.89	0.80	0.07	0.02
	RFCGNN	0.91	0.83	0.05	0.02

To answer this question, we quantified the distance between the counterfactuals identified by RFCGNN and the ground truth counterfactual samples. We then compared this with the distance between the counterfactual samples generated by two other counterfactual methods designed to ensure fair graph counterfactuals, namely NIFTY and GEAR, and the ground truth counterfactual samples. As it is challenging to ascertain the true counterfactual samples for real datasets, we conducted these experiments on synthetic datasets with known ground truth counterfactuals. To assess the disparity between the obtained counterfactuals and the features and structures in the ego graph, we compare the learned counterfactual representation with the ground truth counterfactual representation, the results of which are depicted in Figure 4. The results indicate that the counterfactual samples discovered by RFCGNN are the closest to the ground truth counterfactual samples. Conversely, NIFTY, which merely flips sensitive attributes to obtain its counterfactual samples, ignores the relationship between nodes and their neighbors. As a result, the counterfactual samples it generates exhibit the largest disparity from the real counterfactual samples.Although GEAR utilizes Graph-VAE to generate counterfactuals based on self-perturbation and neighbor perturbation, the absence of supervision results in a larger gap than if true counterfactual samples were identified

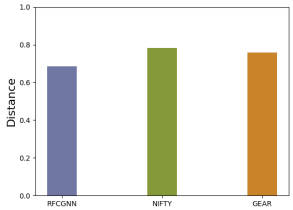

Fig. 4: The distance between identified counterfactual representation and actual ground-truth counterfactual representation.

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

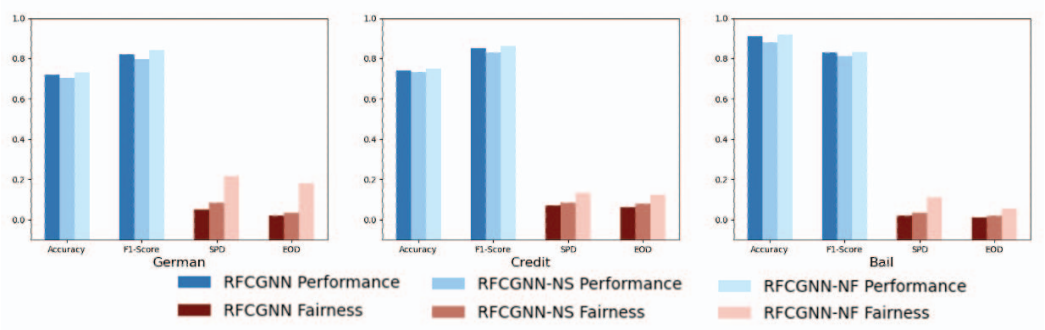

Fig. 5: Ablation study results for RFCGNN, RFCGNN-NS, and RFCGNN-NF.

directly from the observed samples.

RQ3: What is the impact on RFCGNN's performance when individual components are ablated?

To validate the effectiveness of our proposed modules, we conducted an ablation study. Our first analysis examined the significance of the self-paced learning module. For comparison, we removed this module, replacing it with a variant RFCGNN-NS that uses a pre-trained standard GNN to provide pseudo-labels for the nodes in the unlabeled set. The results are depicted in Figure 5. Without the self-paced learning module, RFCGNN-NS suffers in terms of fairness compared to the full RFCGNN model. This inferior performance can be attributed to the label sparsity issue: deprived groups, typically sparser than favored ones in the graph, encounter a higher label acquisition cost during propagation, leading to a greater fairness loss. Further, the scarcity of labels within the deprived group degrades the quality of their counterfactual samples, impacting the overall model performance.

Next, we evaluated the role of the fairness module by removing it to create an RFCGNN-NF variant. As shown in Figure 5, the performance of RFCGNN-NF drops significantly, demonstrating the importance of the fairness module. Without this module, the model fails to eliminate the potential bias inherent in the data.

RQ4: How sensitive is the performance of RFCGNN to variations in its hyper-parameters?

Our model, RFCGNN, has four pivotal hyperparameters, namely α, β, ω, and γ. These control the model's reconstruction performance, the sufficiency regularization's contribution, the fair representation regularization, and label propagation, respectively. Figure 6 illustrates the impact of manipulating these hyperparameters on the model's performance and fairness. For this analysis, we individually varied each hyperparameter through a range from 0 to 10.

When we increase α within a certain range, we observe a modest improvement in performance, albeit at the cost of reduced fairness. Adjusting β and ω upwards results in a trade-off, causing prediction performance to deteriorate while enhancing fairness. As for γ, small increases initially lead to an improvement in both the model's performance and fairness. However, beyond this range, further changes to γ do not significantly impact the model. To summarize, α influences model performance, β and ω guide the representation disentanglement and desensitization, improving fairness, while γ assists the model in better learning minority class nodes.

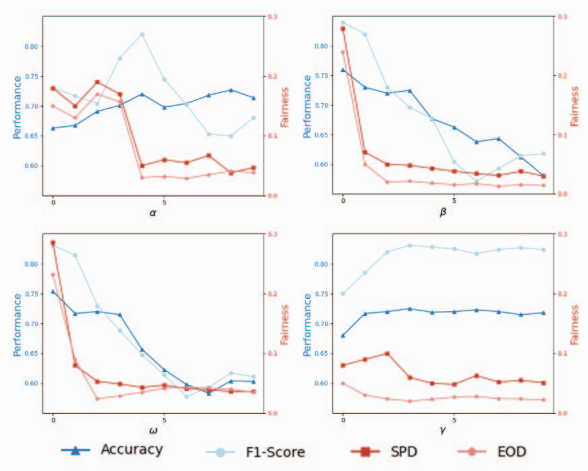

Fig. 6: Exploring hyperparameters study results in the German dataset.

VI. Conclusions

This work is driven by the growing concern over discriminatory behaviors in graph-based decision systems, and aims to achieve fair and accurate predictions. In contrast to existing fair GNN methods, our approach is inspired by causal theory. We propose a novel framework, RFCGNN, to mitigate biases associated with sensitive attributes and labeling processes, offering a unique method to identify real counterfactual samples directly from observed datasets. Extensive experiments demonstrate that RFCGNN can achieve state-of-the-art performance in both synthetic and real-world datasets concerning the prediction-fairness trade-off.

Acknowledgement

This work was supported in part by National Science Foundation (NSF) under Grant No. 2245895 and an NVIDIA GPU Grant.

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.

REFERENCES

[1] S. Zhang, D. Zhou, M. Y. Yildirim, S. Alcorn, J. He, H. Davulcu, and H. Tong, "Hidden: hierarchical dense subgraph detection with application to financial fraud detection," in *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 2017, pp. 570–578.

[2] J. Wu, X. Wang, F. Feng, X. He, L. Chen, J. Lian, and X. Xie, "Self-supervised graph learning for recommendation," in *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 2021, pp. 726–735.

[3] H. Wan, Y. Zhang, J. Zhang, and J. Tang, "Aminer: Search and mining of academic social networks," *Data Intelligence*, vol. 1, no. 1, pp. 58–76, 2019.

[4] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[5] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan, "A survey on bias and fairness in machine learning," *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.

[6] Z. Wang, Y. Zhou, M. Qiu, I. Haque, L. Brown, Y. He, J. Wang, D. Lo, and W. Zhang, "Towards fair machine learning software: Understanding and addressing model bias through counterfactual thinking," *arXiv preprint arXiv:2302.08018*, 2023.

[7] W. Zhang and J. C. Weiss, "Fair decision-making under uncertainty," in *2021 IEEE international conference on data mining (ICDM)*. IEEE, 2021, pp. 886–895.

[8] Z. Wang, C. Wallace, A. Bifet, X. Yao, and W. Zhang, ": Fairness-aware graph generative adversarial networks," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2023, pp. 259–275.

[9] J. Ma, R. Guo, M. Wan, L. Yang, A. Zhang, and J. Li, "Learning fair node representations with graph counterfactual fairness," in *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, 2022, pp. 695–703.

[10] Ö. D. Köse and Y. Shen, "Fairness-aware node representation learning," *arXiv preprint arXiv:2106.05391*, 2021.

[11] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[12] 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.

[13] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018, pp. 335–340.

[14] K. Makhlouf, S. Zhioua, and C. Palamidessi, "Survey on causal-based machine learning fairness notions," *arXiv preprint arXiv:2010.09553*, 2020.

[15] A. Olteanu, C. Castillo, F. Diaz, and E. Kıcıman, "Social data: Biases, methodological pitfalls, and ethical boundaries," *Frontiers in big data*, vol. 2, p. 13, 2019.

[16] 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.

[17] W. Zhang, S. Pan, S. Zhou, T. Walsh, and J. C. Weiss, "Fairness amidst non-iid graph data: Current achievements and future directions," *arXiv preprint arXiv:2202.07170*, 2022.

[18] C. Russell, M. J. Kusner, J. Loftus, and R. Silva, "When worlds collide: integrating different counterfactual assumptions in fairness," *Advances in neural information processing systems*, vol. 30, 2017.

[19] A. Beutel, J. Chen, Z. Zhao, and E. H. Chi, "Data decisions and theoretical implications when adversarially learning fair representations," *arXiv preprint arXiv:1707.00075*, 2017.

[20] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, and P. Lio, "', and yoshua bengio," *Graph attention networks. ArXiv, abs/1710.10903*, vol. 2, no. 6, p. 13, 2018.

[21] S. Bhagat, G. Cormode, and S. Muthukrishnan, "Node classification in social networks," *arXiv preprint arXiv:1101.3291*, 2011.

[22] Y. Sui, X. Wang, J. Wu, M. Lin, X. He, and T.-S. Chua, "Causal attention for interpretable and generalizable graph classification," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 1696–1705.

[23] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe, "Weisfeiler and leman go neural: Higher-order graph neural networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4602–4609.

[24] T. Zhao, G. Liu, D. Wang, W. Yu, and M. Jiang, "Learning from counterfactual links for link prediction," in *International Conference on Machine Learning*. PMLR, 2022, pp. 26 911–26 926.

[25] M. Zitnik and J. Leskovec, "Predicting multicellular function through multi-layer tissue networks," *Bioinformatics*, vol. 33, no. 14, pp. i190–i198, 2017.

[26] H. Yuan, H. Yu, S. Gui, and S. Ji, "Explainability in graph neural networks: A taxonomic survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[27] Z. Jiang, X. Han, C. Fan, Z. Liu, N. Zou, A. Mostafavi, and X. Hu, "Fmp: Toward fair graph message passing against topology bias," *arXiv preprint arXiv:2202.04187*, 2022.

[28] 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.

[29] R. Binns, "On the apparent conflict between individual and group fairness," in *Proceedings of the 2020 conference on fairness, accountability, and transparency*, 2020, pp. 514–524.

[30] W. Zhang and J. C. Weiss, "Longitudinal fairness with censorship," in *proceedings of the AAAI conference on artificial intelligence*, vol. 36, no. 11, 2022, pp. 12 235–12 243.

[31] W. Zhang, T. Hernandez-Boussard, and J. Weiss, "Censored fairness through awareness," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 12, 2023, pp. 14 611–14 619.

[32] N. A. Saxena, W. Zhang, and C. Shahabi, "Missed opportunities in fair ai," in *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 2023, pp. 961–964.

[33] A. Chouldechova, "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments," *Big data*, vol. 5, no. 2, pp. 153–163, 2017.

[34] Z. Wang, N. Saxena, T. Yu, S. Karki, T. Zetty, I. Haque, S. Zhou, D. Kc, I. Stockwell, A. Bifet *et al.*, "Preventing discriminatory decision-making in evolving data streams," *arXiv preprint arXiv:2302.08017*, 2023.

[35] W. Zhang and E. Ntoutsi, "Faht: an adaptive fairness-aware decision tree classifier," *arXiv preprint arXiv:1907.07237*, 2019.

[36] S. Sharifi-Malvajerdi, M. Kearns, and A. Roth, "Average individual fairness: Algorithms, generalization and experiments," *Advances in neural information processing systems*, vol. 32, 2019.

[37] W. Zhang, Z. Wang, J. Kim, C. Cheng, T. Oommen, P. Ravikumar, and J. Weiss, "Individual fairness under uncertainty," in *Frontiers in Artificial Intelligence and Applications*, ser. ECAI 2023, vol. 372. ECAI, 2023, pp. 3042–3049.

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

[39] S. Chiappa, "Path-specific counterfactual fairness," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 7801–7808.

[40] M. J. Kusner, J. Loftus, C. Russell, and R. Silva, "Counterfactual fairness," *Advances in neural information processing systems*, vol. 30, 2017.

[41] Y. Dong, J. Kang, H. Tong, and J. Li, "Individual fairness for graph neural networks: A ranking based approach," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 300–310.

[42] A. Asuncion and D. Newman, "Uci machine learning repository," 2007.

[43] I.-C. Yeh and C.-h. Lien, "The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients," *Expert systems with applications*, vol. 36, no. 2, pp. 2473–2480, 2009.

[44] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" *arXiv preprint arXiv:1810.00826*, 2018.

[45] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[46] Y. Dong, N. Liu, B. Jalaian, and J. Li, "Edits: Modeling and mitigating data bias for graph neural networks," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 1259–1269.

Authorized licensed use limited to: SOUTH CHINA UNIVERSITY OF TECHNOLOGY. Downloaded on June 13,2024 at 11:35:56 UTC from IEEE Xplore. Restrictions apply.