Graph Fairness via Authentic Counterfactuals: Tackling Structural and Causal Challenges

Zichong Wang[1], Zhipeng Yin[1], Fang Liu[2], Zhen Liu[3], Christine Lisetti[1], Rui Yu[4], Shaowei Wang[5], Jun Liu[6], Sukumar Ganapati[1], Shuigeng Zhou[7], and Wenbin Zhang[1*]
[1] Florida International University, Miami, FL, USA
[2] University of Notre Dame, Notre Dame, IN, USA
[3] Guangdong University of Foreign Studies, Guangzhou, China
[4] University of Louisville, Louisville, KY, USA
[5] University of Manitoba, Winnipeg, Manitoba, Canada
[6] Carnegie Mellon University, Pittsburgh, PA, USA
[7] Fudan University, Shanghai, China

ABSTRACT

The extensive use of graph-based Machine Learning (ML) decision-making systems has raised numerous concerns about their potential discrimination, especially in domains with high societal impact. Various fair graph methods have thus been proposed, primarily relying on statistical fairness notions that emphasize sensitive attributes as a primary source of bias, leaving other sources of bias inadequately addressed. Existing works employ counterfactual fairness to tackle this issue from a causal perspective. However, these approaches suffer from two key limitations: they overlook hidden confounders that may affect node features and graph structure, leading to an oversimplification of causality and the inability to generate authentic counterfactual instances; they neglect graph structure bias, resulting in over-correlation of sensitive attributes with node representations. In response, this paper introduces the *Authentic Graph Counterfactual Generator (AGCG)*, a novel framework designed to mitigate graph structure bias through a novel fair message passing technique and to improve counterfactual sample generation by inferring hidden confounders. Comprising four key modules – subgraph selection, fair node aggregation, hidden confounder identification, and counterfactual instance generation – AGCG offers a holistic approach to advancing graph model fairness in multiple dimensions. Empirical studies conducted on both real and synthetic datasets demonstrate the effectiveness and utility of AGCG in promoting fair graph-based decision-making.

1. INTRODUCTION

Graph data is prevalent in real-world scenarios, such as financial markets [54], item recommendations [49], and social networks [35]. Distinguished from tabular data, graph data incorporates both individual node attributes and pertinent structural information, offering an efficient mechanism to represent and analyze complex interrelationships among individuals [61]. Consequently, recent years have witnessed a surge of interest in the development and application of graph algorithms specifically designed for graph data. Among them, graph neural networks (GNNs) have shown great ability in modeling graph-structural data [16, 50], consistently delivering exceptional performance across a diverse range of tasks and applications [41]. Nevertheless, like many ML methodologies, GNNs have been observed to potentially discriminate against certain populations as identified by the *sensitive attribute* (*e.g.,* gender or race), leading to substantial ethical considerations.

To mitigate discrimination in GNNs, existing works primarily leverage statistical fairness notions to address bias in graph representation learning [42, 6, 48]. Their foundation lies in the assumption that bias originates solely from sensitive attributes, aiming to achieve predictions that are statistically equitable across subgroups. However, this strategy largely overlooks the widespread existence of labeling bias, where the labels of the samples are affected by factors unrelated to their determination, such as statistical anomalies [26]. Recent works [53, 46] have thus extended the concept of counterfactual fairness [17] to graphs, seeking to overcome the limitation in the presence of labeling bias by considering the causal relationships between variables. Specifically, this adaptation aims to ensure that nodes and their corresponding counterfactual instances (different versions of the nodes) receive consistent prediction results [48]. For example, in a job recruitment scenario, two candidates with different sensitive attribute values but similar qualifications should have equal hiring opportunities.

The existing counterfactual generating works predominantly generate instances by directly flipping sensitive attributes or perturbing node features. For instance, NIFTY [1] introduces perturbations to sensitive attributes to maximize the similarity between original and altered representations, thereby promoting invariance. Similarly, GEAR [22] employs GraphVAE [43] to minimize the discrepancy between original and counterfactual representations to eliminate the impact of sensitive attributes. Despite these advancements, these methods often produce potentially unauthentic counterfactuals [9, 5]. This is attributed to the fact that counterfactual inference is essentially an unsupervised learning task, and these methods tend to rely on oversimplified causal models that neglect unobserved hidden confounders, which

* Corresponding author
Email: {ziwang, wenbin.zhang}@fiu.edu

affect both the historical choice of treatment and the outcomes, thus preventing the accurate inference of causal effects [40]. For instance, socio-economic status, although unobserved, can influence both the type of medication a patient has access to and the patient's overall health. Without accounting for socio-economic status, it is challenging to isolate the causal effect of medications on health outcomes. Consequently, during counterfactual fairness assessment, it becomes problematic to discern whether a change in a patient's healthcare decision is due to a modification in a sensitive attribute or a shift in a confounder.

Furthermore, existing works on graph counterfactual fairness often overlook the impact of graph structure bias in GNNs [38]. Typically, GNNs employ a uniform message-passing mechanism that aggregates information from neighboring nodes, thereby preserving the topology and node feature information [23]. However, this process can inadvertently amplify existing biases within the graph's structure. In particular, the message-passing approach tends to homogenize the representations of connected nodes. Consequently, nodes connected by intra-group edges, which often exhibit similar features, may become over-represented [7]. In contrast, nodes linked by inter-edges, typically characterized by differing attributes (*i.e.,* high-frequency signals), might be under-represented during this aggregation process [36]. This imbalance often leads to a diminished representation of nodes from diverse sensitivity groups in the final node embedding. Therefore, constructing node edges for counterfactual instances based on node feature similarity often results in nodes with the same sensitive attributes being more closely connected [29]. This practice can inadvertently lead to unintended inter-group isolation and introduce structural bias (*i.e.*, intra- and inter-group edge distribution drift).

To address these limitations, this paper explores the domain of graph counterfactual fairness, with a focus on the potential causal interactions between each sample and its neighboring nodes. In addition, the impact of a sample's hidden confounders on its counterfactual instances, along with the influence of graph structure bias on the generation of these instances, is specifically examined. This area is largely underexplored with unique challenges: **i) Complexity of counterfactual graph data:** Unlike tabular data, which typically follows the principle of being independent and identically distributed (I.I.D.), graph data encompasses both node information and structural interconnections. The intricate nature of these relationships among the nodes implies that creating counterfactual samples requires the generation of features in the corresponding counterfactual scenario but also its interconnections with other nodes. **ii) Identifying hidden confounders:** The key to accurately generating counterfactual instances lies in identifying hidden confounders. However, since hidden confounders are not observable, determining how to accurately identify the hidden confounders of a sample based on its observable attributes is crucial yet challenging for obtaining authentic counterfactual scenarios. **iii) Effective learning inter-group edges:** Disparities in the edge distribution of central nodes can lead to an overrepresentation of neighboring nodes with the same sensitive attributes as the central node during node aggregation. This, in turn, causes an over-correlation of node embeddings with sensitive attributes, posing significant challenges in effectively learning inter-group edges during node aggregation to avoid introducing topological bias.

In response to the aforementioned challenges, this paper proposes a novel framework for graph-based fair ML decision-making. *To the best of our knowledge, this is the first work to mitigate the multi-source bias arising from sensitive attributes, labeling processes, and graph structure in graph-based models by considering both hidden confounders and fair node aggregation.* Specifically, in addition to the causal relationship explored by existing methods, which takes into account the interplay between sensitive attributes, graph structure, and non-sensitive attributes, our proposed causal model further encompasses the presence of hidden confounders, as estimated from the observed node features and graph structure and corresponding influence on them. Subsequently, the graph structure, node features, and the identified hidden confounders are used to learn a counterfactual instance generation function. Additionally, to improve the structural realism of counterfactual instances, an adaptive subgraph extractor is introduced to extend the subgraph by including neighbors that are important to the target node, even if they are far away. Different filter channels with an adaptive encoder are also constructed to discriminately aggregate neighboring information along intra- and inter-edges, which avoids over-correlation of node representations with sensitive attributes, thereby enhancing the quality of node embedding. Finally, the generated counterfactual graphs are employed to ensure prediction consistency across real-world and counterfactual scenarios, achieving graph counterfactual fairness.

The **key contributions** of this paper are: i) We formulate a new graph counterfactual fairness problem that demands concurrent alleviation of algorithmic biases associated with sensitive attributes, labeling processes, and the inherent structure of the graph; ii) We introduce AGCG, a novel framework crafted to attain counterfactual fairness in graphs. By concurrently tackling biases through the identification of hidden confounders and the implementation of fair message passing, it provides a holistic approach to mitigating bias in graphs; and iii) We conduct extensive experiments on three real-world benchmark datasets and a synthetic dataset to demonstrate the superiority of our proposed framework in terms of both bias mitigation and node classification performance.

2. RELATED WORK

2.1 Graph Neural Networks

Graph neural networks have found widespread utility in various tasks involving graph-structured data, such as node classification [33, 16, 3], graph classification [32, 25], and link prediction [62]. Their superior performance is attributed to their ability to represent learning on graphs, with two primary approaches: spectral-based and spatial-based. Specifically, spectral-based approaches, grounded in graph theory, adapt convolution operations to graph data and rely on the graph Laplacian matrix or the adjacency matrix to capture structural information about the graph in the spectral domain [37, 36, 45]. For instance, Graph Convolutional Networks (GCN) simplify graph convolution using a first-order approximation of these operations [16]. In contrast, spatial-based GNNs, like GraphSage [10] and EGNN [19], focus on learning node representations by aggregating infor-

mation from neighboring nodes. Despite the methodological differences, most GNNs involve message passing—a process of pattern extraction and interaction modeling within each layer. However, a major challenge for this framework is how to effectively aggregate node information from neighbors with different sensitive attributes. Existing uniform aggregation leads to suppression of information from neighboring nodes with different sensitive attributes, introducing structural biases that, in turn, affect the fairness and performance of downstream tasks.

2.2 Fairness on Graphs

Fairness in graphs has received intensive attention. Most of the existing methods are based on statistical fairness notations such as group fairness [39, 11, 47] and individual fairness [31, 28, 43]. Specifically, group fairness evaluates whether the outcome statistics of the classifiers are similar across different subgroups [34], while individual fairness ensures that similar individuals receive similar probability distributions over class labels [8]. Despite their great success, they inadequately address label bias. To this end, counterfactual fairness [17] leverages the causal theory to eliminate the root bias. Existing works on graph counterfactual fairness either generate counterfactual instances [58] or identify potential counterfactual instances within input dataset [48]. However, the former relies on oversimplified causal models that neglect unobservable hidden confounders, failing to capture authentic counterfactual scenarios [27]. Meanwhile, the latter incorrectly assumes the presence of authentic counterfactual instances within the input data, an assumption that may not hold true. Consequently, both approaches struggle to reflect authentic counterfactual scenarios.

To jointly address these challenges, we aim to generate authentic counterfactual instances by acknowledging the existence of hidden confounders, identifying them with a variational inference approach with Gaussian mixture priors, and incorporating the information when learning the generating functions of counterfactual instances. Furthermore, our approach addresses the root cause of graph structure bias, enhances fairness in the node aggregation process, and improves the quality of node embeddings with minimal impact on overall model performance.

3. NOTATION

Consider an undirected and unweighted input graph with n nodes as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges such that $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and $\mathbf{X}$ is the set of node features with $x_i \in \mathbb{R}^{1\times D}$ representing the features of individual node i. Each node v_i has a binary sensitive attribute s_i, indicating whether node v_i belongs to a deprived group ($s_i = 0$) or favored group ($s_i = 1$), which is part of the feature set $\mathbf{X}$. We use $S \in \{0,1\}^{N\times 1}$ ($S \in \mathbf{X}$) to denote the vector representing the sensitive attributes of nodes. The adjacency matrix of the graph $\mathcal{G}$ is denoted as $A \in \{0,1\}^{n\times n}$, where $A_{i,j} = 1$ if there is an edge between nodes v_i and v_j, and 0 otherwise. An edge $A_{i,j}$ is classified as an intra-group edge if nodes v_i and v_j share the same sensitive attribute value, and as an inter-group edge otherwise. We let v_{syn}, and s_{syn} denote the generated node, its sensitive attribute, respectively. Additionally, let $C = [C_1, \ldots, C_n]$ denote the matrix of hidden confounders, where each $C_i \in \mathbb{R}^p$ represents the confounders for node v_i. Without loss of generality, we use $\mathcal{L} = \{v_1, v_2, \ldots, v_L\}$ to signify the set of $\mathcal{L}$ labeled vertices, accompanied by their observed labels $Y = \{y_1, \ldots, y_L\}$, with y_i denoting the ground-truth label of vertex v_i. We also use $\mathcal{U} = \{v_{L+1}, v_{L+2}, \ldots, v_{L+U}\}$ representing the set of $\mathcal{U}$ unlabeled vertices, and the predicted labels are $\hat{Y} = \{\hat{y}_1, \ldots, \hat{y}_L\}$. Also note that $\mathcal{L} \bigcup \mathcal{U} = \mathcal{V}$.

4. METHODOLOGY

4.1 Causal Model

Figure 2 depicts the causal model, which serves as the foundation of AGCG for fair counterfactual decision-making. *To the best of our knowledge, this is the first causal model that delves into the causal relationships between hidden confounders (C) and observable attributes: sensitive attributes (S), node features (X), graph structure (A), and ground-truth label (Y) in the realm of counterfactual fairness.* In the proposed model, each connection represents a deterministic causal link, indicating the direct influence of one variable on another. Through this framework, AGCG can discern potential modifications that would occur in the counterfactual world under different conditions. Below, we delineate the rationale and explanations that underpin the causal model.

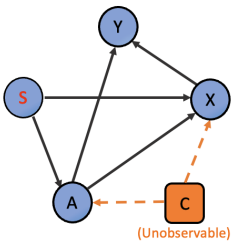

Figure 2: The causal model of AGCG.

- $A \leftarrow C \rightarrow X$: The hidden confounder C has implications for the graph structure A and node features X but does not directly affect sensitive attribute, nor can it be impacted by the graph structure and node features. For instance, a person's "bad temper" might influence his/her "blood pressure" and deteriorate his/her social relationships with others. However, it cannot change a person's "gender". Note that C represents unobservable features and might not always correspond to tangible entities in the real world. In addition, a causal path from C to ground-truth label Y is hypothesized to be mediated through observable variables (*i.e.,* A and X).

- $A \leftarrow S \rightarrow X$: Since the sensitive attribute S is typically determined at birth, there is no parent variable in the causal graph. Instead, S can only serve as the cause of other variables, which in turn influence the node's features X and graph structure A. For instance, the sensitive attribute "gender" cannot be caused by other features such as "height", whereas "gender" can influence "height". Similarly, on social networks, the "gender" of a person might skew their connections toward similarly gendered individuals, while these connections cannot change a person's "gender". Notably, S does not affect the hidden confounder C, *e.g.*, a person's "gender" does not affect their "health".

- $Y \leftarrow A \rightarrow X$: The graph structure A has implications for both node features X and ground-truth label Y, *i.e.*, potential for changes in one node to impact another. For

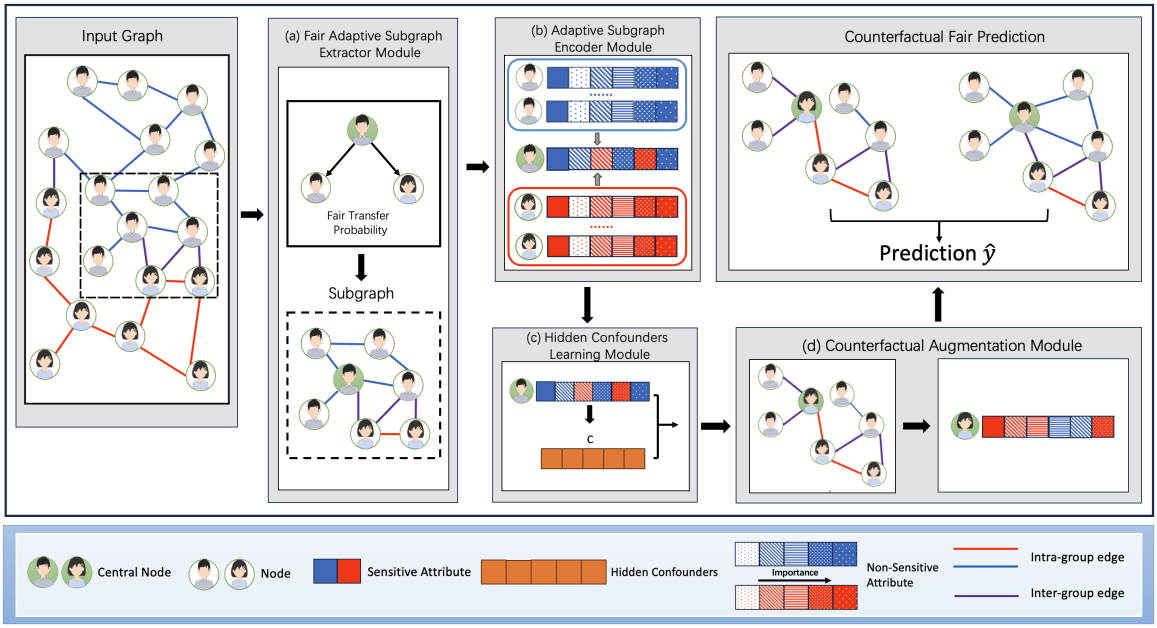

Figure 1: Overview of the proposed AGCG framework. For each node, the framework first extracts its contextual subgraph. It then fair adaptively aggregates information from both intra- and inter-edges, infer the hidden confounders from the observed data and generates counterfactual instances using both the observed data and these hidden confounders.

example, if all of a person's friends have watched a particular movie, that person is also likely to watch it, thereby changing his/her movie-watching history.

- $X \to Y$: The node features X has impacts for the ground-truth label Y. Notably, a causal path from X to A would have similar total causal effects as a causal path from A to X. Therefore, for computational efficiency, we assume that there is no causal path from X to A.

4.2 AGCG: In a Nutshell

Expanding upon the established causal model, the proposed AGCG framework, comprising four modules, is illustrated in Figure 1: (a) The *Fair Adaptive Subgraph Extractor Module* (Section 4.3), which adaptively identifies contextual subgraphs relevant to each node; (b) The *Adaptive Subgraph Encoder Module* (Section 4.4), tailored to differentially aggregate information from intra-group and inter-group edges linked to each node; and (c) The *Hidden Confounders Learning Module* (Section 4.5), dedicated to inferring hidden confounders for each node based on its observed features and structure; (d) The *Counterfactual Augmentation Module* (Section 4.6), responsible for creating counterfactual instances by integrating observed graph data with the inferred hidden confounders. Subsequent sections will delve into the details of each module.

4.3 Fair Adaptive Subgraph Extractor Module

Learning causal models directly on large-scale graph data (*e.g.*, social networks) can be computationally expensive. To mitigate this complexity, a common strategy is to extract a local subgraph for each node, based on the assumption that a node is predominantly influenced by its immediate neighborhood [14]. Drawing upon previous work [39], the proposed subgraph extraction module operates under the premise that each node v_i depends minimally on nodes outside a certain "context subgraph". This context subgraph aims to retain essential structural and relational information relevant to v_i.

In addition, unlike approaches that restrict the neighborhood size (*e.g.*, to 1-hop neighbors), our method leverages importance scores ($ImpS$) to identify influential nodes, regardless of their distance. As depicted in Figure 1 (a), for a given central node (highlighted in green), we extract a context subgraph (shown within the dashed black rectangle) composed of the top-k most influential neighbor nodes based on $ImpS$. In doing so, we broaden the local neighborhood beyond immediate neighbors, incorporating distant yet informative nodes. This enhances both representation learning and counterfactual data augmentation.

To compute $ImpS$, we first construct a normalized adjacency matrix $\overline{A} = AD^{-1}$, where D is a diagonal degree matrix with entries $D_{i,i} = \sum_j A_{i,j}$. However, this standard normalization does not account for fairness concerns, as it fails to consider how nodes sharing certain sensitive attributes often form densely connected substructures. These substructures can distort transition probabilities, amplifying subgroup disparities within the extracted subgraphs [13]. To counteract this bias, a fairness constraint on the transition probabilities is enforced. Specifically, neighboring nodes are classified by their sensitive attributes, ensuring that the aggregate selection probabilities are balanced across these attributes, and this is formally imposed as:

$$\sum(P_{v_a}|\overline{A}_{a,j} = 1, s_a \in S_d) = \sum(P_{v_b}|\overline{A}_{b,j} = 1, s_b \in S_f) \quad (1)$$

where P_{v_a} and P_{v_b} are the transition probabilities to neighboring nodes in deprived (S_d) and favored (S_f) groups, respectively. With this fairness-aware normalization, $ImpS$ is then computed as:

$$ImpS = \alpha(I - (1-\alpha)\overline{A})^{-1} \tag{2}$$

where I is the identity matrix and $\alpha \in [0,1]$ controls the restart probability from the central node. Each entry $ImpS_{i,j}$ measures the importance of node v_j to node v_i, and $ImpS_{i,:}$ denotes the importance vector for node v_i. This computation is performed as a pre-processing step, thus not incurring additional overhead during model training. Armed with these importance scores, our Adaptive Subgraph Extraction module selects the top-k high-$ImpS$ nodes for each central node v_i to form its context subgraph $\mathcal{G}_{v_i}$.

4.4 Adaptive Subgraph Encoder Module

After extracting the subgraphs, AGCG aggregates information from each central node's neighbors to obtain final node embeddings. Existing approaches [4, 21] often apply uniform message-passing over both intra- and inter-group edges without differentiating among distinct information frequencies (*e.g.*, low-frequency, high-frequency, and identity information). This uniform treatment may cause the learning process to be dominated by the majority edge type (*i.e.*, intra-group edges) while neglecting critical signals from less-represented edges (*i.e.*, inter-group edges).

To address this issue, we propose a new subgraph encoder that considers the disproportion in intra- and inter-group edge distributions. By distinctly handling information derived from these edges, our encoder reduces bias and ensures that embeddings adequately represent diverse subgroups. For example, as shown in Figure 1(b), the encoder separately aggregates information from intra-group edges (in the blue box) and inter-group edges (in the red box). This approach enriches the final embedding for the central node with valuable signals from nodes having different sensitive attributes, mitigating bias introduced by skewed edge distributions. Specifically, we introduce three types of learnable weights to handle different frequency components: i) Low-frequency weight (ω_L). We capture commonalities between the central node and its neighbors by concatenating their transformed features. Mathematically denoted as $\omega_L(v_i, k_i) = \sigma(\mathbf{u}_L^\top(W_L z_L(v_i, k_i)))$, where $\mathbf{u}_L$ is a learnable vector. ii) High-frequency weight (ω_H). To highlight differences between neighbors, especially those with distinct sensitive attributes, we incorporate a negative sign on the neighbor's features before transformation. Mathematically, it represent as $\omega_H(v_i, k_i) = \sigma\big(\mathbf{u}_H^\top(W_H(-h_{k_i}^{(l-1)}))\big)$. iii) Identity weight ($\omega_I$). To preserve the central node's inherent characteristics, we define: $\omega_I(v_i, k_i) = \sigma(\mathbf{u}_I^\top(W_I h_{v_i}^{(l-1)}))$. Here, $\sigma(\cdot)$ is the sigmoid activation function, W_L, W_H, W_I are layer-specific transformation matrices for different frequency components, and $\mathbf{u}_L, \mathbf{u}_H, \mathbf{u}_I$ are learnable parameter vectors.

To effectively integrate these weights, we normalize them across the three information types for each node pair (v_i, k_i): $\hat{\omega}_{(v_i,k_i)} = \big[\overline{\omega}_L(v_i,k_i), \overline{\omega}_H(v_i,k_i), \overline{\omega}_I(v_i,k_i)\big]$, where $\overline{\omega}_{a\in\{L,H,I\},(v_i,k_i)}$ is calculated as: $\overline{\omega}_a(v_i,k_i) = \mathrm{softmax}\left(\omega_a(v_i,k_i)\right)$. The obtained weighting vector $\hat{\omega}_{(v_i,k_i)}$ is used to aggregate the multi-frequency information from neighboring nodes to compute the central node embedding:

$$\begin{aligned} h_{v_i}^l =& \mathrm{UPD}_k^l\big(\pi h_{v_i}^{l-1}, \mathrm{AGG}_{k_j\in\mathcal{G}_{v_i}} \\ & \big(\hat{\omega}_{(v_i,k_j)}^{(l)} \mathrm{ReLU}\left(W_R[W_L h_{k_j}^{l-1}, W_H h_{k_j}^{l-1} W_I h_{k_j}^{l-1}]\right)\big)\big) \end{aligned} \tag{3}$$

where π is a hyperparameter, $W_R \in \mathbb{R}^{d_l \times 3d_l}$ denotes the projected matrix, integrating the embeddings from layer $l-1$, and $h_{v_i}^l$ denotes the aggregated neighboring embedding of node $k_i \in \mathcal{G}_{v_i}$ after superimposing the l layer encoder.

To effectively train a fair adaptive subgraph encoder module, we train it with an adjacency matrix reconstruction task. Considering the sparsity of positive edges (*e.g.,* existing edges), we also adopt negative sampling (*e.g.*, non-existing edges) to train our module. To ensure the number of positive samples ($\mathcal{E}^+$) is the same as negative samples ($\mathcal{E}^-$), we randomly choose $|\mathcal{E}^+|$ negative edges from the total negative edges as negative samples. The reconstruction loss $\mathcal{L}_{rec}$ is calculated as follows:

$$\mathcal{L}_{rec} = \frac{1}{|\mathcal{E}^+| + |\mathcal{E}^-|} \sum_{e_{ij}\in\mathcal{E}} L(e_{ij}, \hat{e}_{ij}) \tag{4}$$

where $\hat{e}_{ij}$ and e_{ij} are the predicted and observed edge of input graph $\mathcal{G}$, respectively. This approach effectively trains the model to distinguish between existing and non-existing links, enhancing its ability to accurately reconstruct the graph structure.

4.5 Hidden Confounders Learning Module

In this section, we discuss how AGCG generates authentic counterfactual instances to achieve graph counterfactual fairness. According to the causal analysis in Section 4.1, generating these instances relies on accurately approximating the joint distribution $P(C, A, X, S) = P(C|X, A, S)P(A, X, S)$. However, this task is complicated by the unobservability of hidden confounders C and the computational infeasibility of directly calculating the marginal likelihood $P(X, A, S)$ due to the need to integrate C. To this end, we optimize the Evidence Lower Bound [15] (ELBO) related to the marginal log-likelihood of the observable graph data (*e.g.*, A, X, S), which allows us to effectively approximate and recover the joint distribution $P(C, A, X, S)$, thus ensuring the authenticity of the counterfactual instances generated, as demonstrated in Equation 5:

$$\log P(A, X|S) \geq \mathbb{E}_{Q(C|A,X,S)}[\log P(C, A, X, S)] - \mathbb{E}_{Q(C|A,X,S)}[\log Q(C|A, X, S)] \tag{5}$$

where $Q(C|A, X, S)$ denotes the variational distribution that uses a parametric family of distributions to approximate the intractable posterior distribution $P(C|A, X, S)$. This strategy allows us to sample C from its posterior given the observable variables (*e.g.*, A, X, S), as formalized in Equation 6:

$$P(C|X, A, S) = \frac{P(X, A, S|C)P(C)}{P(X, A, S)} \tag{6}$$

Building upon this approximation framework, we model the joint distribution $P(C, A, X, S)$ consistently with our pro-

posed causal model (as shown in Figure 2), as outlined in Equation 7:

$$P(C, A, X, S) = P(C)P(S)P(A|C, S)P(X|A, C, S) \tag{7}$$

where $P(A|C, S)$ and $P(X|A, C, S)$ are the graph structure generation function G_A and the node features generation function G_X, respectively, to be detailed in Section 4.6. In addition, the generative and inference model parameters are learned simultaneously by maximizing the ELBO.

However, this modeling strategy relies on the assumption that the VAE model is identifiable, a premise that has not been fully established [20, 24]. This is attributed to the fact that multiple parameter sets can yield models with identical marginal data and prior distributions, yet differ significantly in the hidden confounder C. Consequently, obtaining the true joint distribution $P(C, A, X, S)$ only using VAE is not feasible. To ensure the model in Equation 7 is identifiable, we specify a Gaussian mixture prior to the hidden confounder c_i associated with v_i. To achieve identifiability and capture more complex latent patterns, we impose a mixture of Gaussians prior on the hidden confounder c_i. Specifically, a discrete random variable selects one among a finite set of candidate Gaussian components, each characterized by its own mean vector and covariance matrix. The relative likelihood of choosing each component is governed by a set of mixing proportions that sum to one. By marginalizing over these discrete assignments, the effective prior emerges as a weighted combination of multiple Gaussian densities. By employing this mixture prior, the latent space can accommodate multiple modes and subtle variations in the underlying data, helping to alleviate identifiability issues. This setup thus not only supports a richer and more nuanced characterization of the hidden confounders but also provides a structured probabilistic foundation for improved variational inference. Building on this structured prior, we define a categorical distribution to describe the allocation of weights among the Gaussian components, facilitating the necessary probabilistic framework for effective variational inference, where probabilities of the individual components are formulated using a categorical distribution. Moreover, let T be the vector containing the component values for all nodes within the graph, and the variational distribution $Q(C, T|A, X)$ can be factorized as:

$$Q(C, T|A, X, S) = Q(C|A, X, S)Q(T|A, X, S) \tag{8}$$

Building on this, the updated ELBO of our framework can be formally described as:

$$\begin{aligned}\log P(A, X|S) &= \log \int\int P(C)P(S)P(A|C, S)P(X|A, C, S)\, dC\, dT \\ &\geq \mathbb{E}_{Q(C,T|A,X,S)}\left[\log \frac{P(A, X, C, T|S)}{Q(C, T|A, X, S)}\right]\end{aligned} \tag{9}$$

where the values of $\log P(A, X|S)$ correlate positively with the reality of both the graph structure and node features, while $\mathbb{E}_{Q(C,T|A,X,S)}$ denotes the expectation with respect to the variational distribution Q. Furthermore, $P(A, X, C, T|S)$ represents the joint distribution between the observed data, while $P(C, T|A, X, S)$ denotes the posterior distribution of the hidden confounders.

Using the factorization of the variational distribution, the updated ELBO of our framework can be formally described as:

$$\log P(A, X|S) \geq \mathbb{E}_{Q(C,T|A,X,S)}\left[\log \frac{P(T)P(C|T)P(A|C, S)P(X|A, C, S)}{Q(C, T|A, X, S)}\right] \tag{10}$$

4.6 Counterfactual Augmentation Module

With hidden confounders identified, AGCG proceeds to generate counterfactual instance for each node v_i, which is subsequently used to train downstream classifiers to achieve graph counterfactual fairness. As shown in Figure 1 (d), two generating functions are utilized to generate the graph structure A (left), and node features X (right) of the counterfactual instance, respectively. Specifically, given an observed graph $\mathcal{G} = \{A, X, S\}$, the sensitive attributes S' of each node v_i are flipped (*i.e.,* $S'_i = \neg S_i$), then utilized along with the obtained hidden confounder C to generate the adjacency matrix A' according to $G_A(C, S')$, representing the topology of the counterfactual subgraph (the ego graph of the counterfactual counterpart of v_i) as follows:

$$A'(v_{syn}, v_j) = \frac{1}{1 + e^{-C_{v_{syn}} \cdot C_{v_j}^T}} \tag{11}$$

where $C_{v_{syn}}$ and C_{v_j} represent the hidden confounder of a generated counterfactual instance and of other nodes within the observed graph $\mathcal{G}$, respectively. In addition, to enhance genuineness, real world graph topology distributions are incorporated. Specifically, nodes sharing the same sensitive attribute value (*i.e.,* within the same subgroup) are more likely to form connections. To reflect this real-world phenomenon, two thresholds are established: η_1 for nodes with the same sensitive attribute value and η_2 for nodes with different sensitive attribute values. These thresholds represent the probability that a connection exists between pairs of nodes, depending on whether they share sensitive attribute value, with the exact values to be fine-tuned according to the dataset's distribution. This strategy, informed by the observed input graph topology, aims to ensure that the synthesized graph structures reflect realistic connectivity patterns and avoid overly sparse connections between nodes with differing sensitive attributes, thereby improving the realism and fairness of the counterfactual graph structure.

In terms of generating node features $X' = G_X(C, A', S')$, the process is based on:

$$X'^{(q)}_i = G_{X1}(c_i, s_i, X'^{(q-1)}_i) + \tag{12}$$

$$AGG_{v_j \in \mathcal{G}_i^{(q)}}(G_{X2}(c_j, s_j, X'^{(q-1)}_j)) \tag{13}$$

where $X'^{(q-1)}_i$ denote the $(q-1)^{th}$ layer feature of node v_i, and $AGG(\cdot)$ denotes an aggregation function that maps the information from all neighboring nodes to a single vector. In addition, G_{X1} and G_{X2} are two piecewise affine transformation functions, *e.g.,* multilayer perceptrons with leaky $ReLU$ activations.

Building on these generated authentic counterfactual instances, our classifier is then trained with real samples and their counterfactual counterparts to ensure consistent predicted labels for both. The corresponding loss function $\mathcal{L}_{fair}$ is denoted as:

$$\mathcal{L}_{fair} = -\frac{1}{N}\sum_{i=1}^{N}[\hat{y}_i \log(\hat{y}'_i) + (1-\hat{y}_i)\log(1-\hat{y}'_i)] \quad (14)$$

where $\hat{y}_i$ is the predicted label of the real sample, and $\hat{y}'_i$ is for the counterfactual sample. The fair loss function minimizes the prediction discrepancy between the predictions for real and counterfactual instances, thereby promoting fairness.

To maintain the utility of AGCG, we adopt the cross-entropy loss as the utility loss $\mathcal{L}_{utility}$, defined as:

$$\mathcal{L}_{utility} = \frac{1}{|\mathcal{V}_L|}\sum_{v_i \in \mathcal{V}_L} -(y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)) \quad (15)$$

The overall objective function is ultimately derived from all the loss functions described above.

5. EXPERIMENT

5.1 Datasets

Experiments are conducted on three real-world datasets and a synthetic dataset. For the real-world datasets: i) The **German** dataset [2] contains credit information of clients at a German bank. Each node represents a client, and each edge denotes the similarity between two clients' credit accounts. The sensitive attribute is the clients' gender, aiming to classify clients into good versus bad credit risks. ii) The **Credit** dataset [52] comprises individuals' default payment information, where each node signifies an individual, and edges denotes the similarity in their expenditure and payment patterns. The sensitive attribute is age, with the objective of predicting whether an individual's default mode of payment is via credit card. iii) The **Bail** dataset [1] presents data related to defendants granted bail in U.S. state courts. Each node corresponds to a defendant, and an edge connecting two nodes signifies similarities in their criminal records and demographic details. The race of the defendants is used as the sensitive attribute, with the goal of classifying defendants into two categories: those suitable for bail and those who are not.

As real-world datasets lack ground-truth counterfactuals, a synthetic dataset is constructed using the proposed causal model. This gives us accurate counterfactuals for each node, enabling precise assessment of generated instances. In our setup, we consider the binary sensitive attributes and labels, which are generated based on a Bernoulli distribution, *i.e.,* $s_i \sim Bernoulli(p_s)$. Next, we begin by drawing latent factors C from a Gaussian mixture model with Z elements. For each node, we use G_X and G_A to generate node feature X and adjacency matrix A. Node labels are generated in the same way as node features. This way allows us to manipulate various parameters, including the sensitive attribute probability, label probability, and feature dimensions. Table 1 provides detailed statistics of these four datasets.

5.2 Baselines

To assess AGCG, we compare it against eight state-of-the-art node classification methods, categorized into three groups: i) Vanilla graph model: **GCN** [16], **GraphSAGE** [10], and **GIN** [51]. ii) Fair Node Classification Methods: **Graphair** [12] aims to use adversarial learning to automatically produce fair graph data that can trick the discriminator. **FairAGG** [63] implements a fair aggregation scheme based on the Shapley value to ensure group fairness. iii) Graph Counterfactual Fairness Methods:**NIFTY** [1] create counterfactuals by introducing perturbations to sensitive attributes, thereby enhancing model fairness. **RFCGNN** [44] learns a fair node representation by identifying counterfactual instances and sensitive attribute-related information masking, and **FDGNN** [39] utilizes counterfactual samples to learn disentangled node representation to mitigate the multi-source biases.

Table 1: Summary of the datasets used in the experiments.

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

5.3 Evaluation Metrics

Both fairness and predictive performance are evaluated, with Statistical Parity Differences (SPD) [18] and Equal Opportunity Differences (EOD) [11], with values close to zero indicating better fairness. We utilize AUC and F1-score to measure the performance on node classification tasks, with higher values indicating better performance.

5.4 Experiment results

Comparison Study. Table 2 presents the performance of node classification and fairness, including the standard deviation from 10 experiments, along with the average results. As can be seen, AGCG is affirmed by the empirical results as highly effective. Specifically, AGCG consistently achieves top rankings for fairness metrics across all datasets, when compared with other baseline methods. From the perspective of model utility, AGCG demonstrates comparable results in the F1-score, while the AUC is higher than some baselines. AGCG's advantage stems from its accurate causal model, which accounts for hidden confounders, leading to the generation of authentic counterfactual scenarios that improve the model's graph counterfactual fairness. Furthermore, AGCG effectively mitigates graph structure bias, reducing the likelihood of node embeddings being overly influenced by sensitive attributes, and efficiently leverages key information from neighbors with differing sensitive attributes. Overall, AGCG exhibits good performance in balancing the trade-off between prediction accuracy and fairness.

Ablation Study. To evaluate the effectiveness of the individual components in AGCG, an ablation study was conducted. Initially, the significance of the adaptive subgraph encoder model was examined. For comparison, this component was removed and replaced with the AGCG-NAE variant, utilizing a standard encoder, such as performing uni-

Table 2: Predictive and fairness performance for AGCG and baselines across real-world datasets and synthetic datasets.

Dataset	Methods / Metrics	Vanilla Methods			Fair Node Classification Methods		Graph Counterfactual Fairness Methods			
		GCN	GraphSAGE	GIN	Graphair	FairAGG	NIFTY	RFCGNN	FDGNN	**AGCG**
German	**AUC** (↑)	0.654 ± 0.015	**0.781** ± 0.008	0.734 ± 0.012	0.718± 0.054	0.704 ± 0.020	0.736 ± 0.041	0.747 ± 0.029	**0.781** ± 0.022	0.744 ± 0.048
	F1-Score (↑)	0.786 ± 0.012	0.817 ± 0.019	0.812 ± 0.015	0.813 ± 0.012	0.781 ± 0.014	0.792 ± 0.019	0.823 ± 0.012	**0.837** ± 0.021	0.828 ± 0.037
	SPD(↓)	0.364 ± 0.052	0.231 ± 0.058	0.148 ± 0.046	0.084 ± 0.073	0.063 ± 0.047	0.077 ± 0.028	0.067 ± 0.017	0.058 ± 0.010	**0.057** ± 0.010
	EOD(↓)	0.312 ± 0.041	0.157 ± 0.056	0.091 ± 0.037	0.058 ± 0.023	0.036 ± 0.038	0.049 ± 0.023	0.041 ± 0.016	0.024 ± 0.009	**0.021** ± 0.017
Credit	**AUC** (↑)	0.707 ± 0.017	**0.767** ± 0.013	0.728 ± 0.013	0.758 ± 0.047	0.721 ± 0.022	0.727 ± 0.024	0.743 ± 0.033	0.747 ± 0.031	0.734 ± 0.022
	F1-Score (↑)	0.835 ± 0.028	0.859 ± 0.011	0.809 ± 0.018	0.728 ± 0.072	0.747 ± 0.042	0.806 ± 0.012	0.849 ± 0.049	**0.861** ± 0.048	0.859 ± 0.031
	SPD(↓)	0.108 ± 0.035	0.113 ± 0.037	0.132 ± 0.037	0.085 ± 0.034	0.074 ± 0.036	0.094 ± 0.017	0.074 ± 0.047	**0.056** ± 0.024	0.063 ± 0.024
	EOD(↓)	0.096 ± 0.035	0.124 ± 0.047	0.128 ± 0.047	0.088 ± 0.035	0.056 ± 0.021	0.113 ± 0.027	0.064 ± 0.016	0.047 ± 0.016	**0.043** ± 0.013
Bail	**AUC** (↑)	0.871 ± 0.019	0.894 ± 0.021	0.768 ± 0.067	0.822 ± 0.023	0.803 ± 0.016	0.796 ± 0.008	**0.896** ± 0.017	0.894 ± 0.013	0.866 ± 0.024
	F1-Score (↑)	0.784 ± 0.022	0.793 ± 0.031	0.658 ± 0.088	0.763 ± 0.038	0.743 ± 0.024	0.674 ± 0.062	**0.802** ± 0.032	0.785 ± 0.022	0.768 ± 0.057
	SPD(↓)	0.093± 0.015	0.086 ± 0.035	0.072 ± 0.037	0.051 ± 0.033	0.047 ± 0.035	0.035 ± 0.037	0.031 ± 0.013	0.025 ± 0.011	**0.023** ± 0.018
	EOD(↓)	0.044 ± 0.015	0.041 ± 0.022	0.043 ± 0.027	0.045 ± 0.033	0.036 ± 0.024	0.028 ± 0.023	0.024 ± 0.016	0.020 ± 0.014	**0.018** ± 0.008
Synthetic	**AUC** (↑)	0.653 ± 0.013	**0.705** ± 0.017	0.693 ± 0.015	0.662 ± 0.029	0.657 ± 0.024	0.702 ± 0.041	0.663 ± 0.030	0.695 ± 0.037	0.703 ± 0.033
	F1-Score (↑)	0.657 ± 0.024	0.685 ± 0.019	0.673 ± 0.024	0.676 ± 0.028	0.631 ± 0.032	0.713 ± 0.042	0.689 ± 0.032	0.724 ± 0.047	**0.727** ± 0.050
	SPD(↓)	0.146 ± 0.035	0.138 ± 0.042	0.248 ± 0.055	0.054 ± 0.023	0.040 ± 0.027	0.045 ± 0.024	0.061 ± 0.031	0.041 ± 0.021	**0.032** ± 0.011
	EOD(↓)	0.128 ± 0.032	0.114 ± 0.027	0.183 ± 0.048	0.022 ± 0.033	0.028 ± 0.037	0.038 ± 0.011	0.031 ± 0.017	0.023 ± 0.019	**0.018** ± 0.012

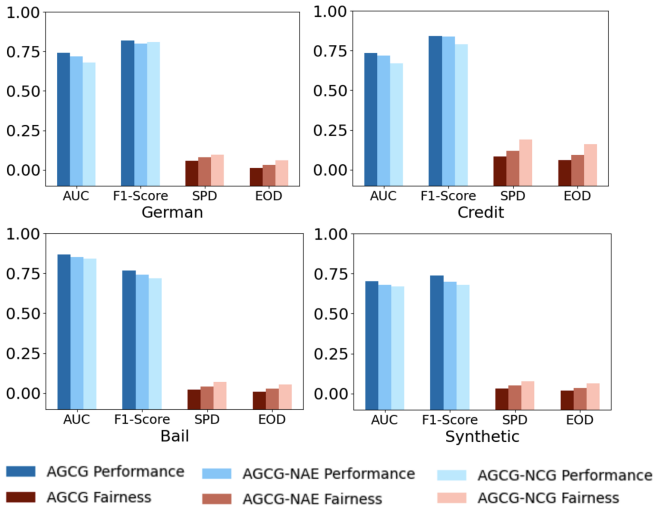

Figure 4: Ablation study results for AGCG, AGCG-NAE, and AGCG-NCG.

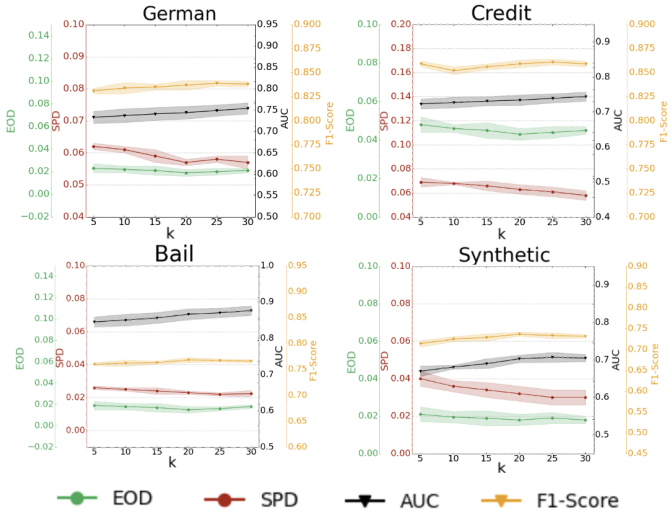

Figure 5: Parameter study on the choice of k-value.

form messaging for both intra- and inter-group edges. As depicted in Figure 4, the fairness of AGCG-NAE notably declined. This decrease is attributed to the model's inability to adequately learn information from neighboring nodes with differing sensitive attributes during the aggregation process, thus introducing structural bias. Additionally, AGCG-NAE can result in the over association of the node representations with sensitive attributes, degradation of the quality of the generated counterfactual instances, and, consequently, model performance. Subsequently, the importance of the counterfactual augmentation model was evaluated by creating an AGCG-NCG variant in its absence. As shown in Figure 4, AGCG-NCG also exhibited a significant drop in fairness performance, underscoring the critical role of the fairness module in mitigating potential biases.

Effect of Different k Values. In the experiments, we evaluated the impact of varying subgraph sizes k, in $\{5, 10, 15, 20, 25, 30\}$, while keeping all other training factors constant. The classification and fairness performance on all datasets are depicted in Figure 5. It is observed that the model achieves better fairness with relatively larger subgraph sizes. Specifically, the model exhibits more substantial fairness enhancements as the subgraph size increases up to 20. However, this improvement becomes less significant when the subgraph size exceeds 20. This is because nodes with $ImpS$ scores ranked after the top 20 hold limited importance to the center node. Consequently, expanding the subgraph beyond 20 nodes yields negligible gains in classification and fairness performance.

6. CONCLUSION

This paper introduces AGCG, a novel graph counterfactual fairness framework designed to enhance the fairness of GNNs. AGCG addresses a critical gap in existing graph counterfactual fairness works, *i.e.,* oversimplified causal models that overlook hidden confounders. Furthermore, by explicitly considering and mitigating the effects of graph structural biases, AGCG ensures consistent representation of different subgroups in node embeddings. The AGCG framework achieves graph counterfactual fairness by simultaneously learning from the original sample and its corresponding authentic counterfactual sample. Experimental evaluations, performed on both synthetic and real-world graph data, substantiate the efficacy of our proposed method in maintaining superior prediction performance while enhancing fairness. This work provides a new perspective on achieving counterfactual fairness in graph data, contributing to the ongoing development of fair GNNs.

Acknowledgement

This work was supported in part by the National Science Foundation (NSF) under Grant No. 2245895.

7. REFERENCES

[1] Chirag Agarwal, Himabindu Lakkaraju, and Marinka Zitnik. "Towards a unified framework for fair and stable graph representation learning". In: *Uncertainty in Artificial Intelligence*. PMLR. 2021, pp. 2114–2124.

[2] Arthur Asuncion and David Newman. *UCI machine learning repository*. 2007.

[3] Smriti Bhagat, Graham Cormode, and S Muthukrishnan. "Node classification in social networks". In: *arXiv preprint arXiv:1101.3291* (2011).

[4] Deyu Bo et al. "Beyond low-frequency information in graph convolutional networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 5. 2021, pp. 3950–3957.

[5] Sribala Vidyadhari Chinta et al. "FairAIED: Navigating fairness, bias, and ethics in educational AI applications". In: *arXiv preprint arXiv:2407.18745* (2024).

[6] Zhibo Chu, Zichong Wang, and Wenbin Zhang. "Fairness in Large Language Models: A Taxonomic Survey". In: *ACM SIGKDD Explorations Newsletter, 2024* (2024), pp. 34–48.

[7] Enyan Dai and Suhang 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.

[8] Cynthia Dwork et al. "Fairness through awareness". In: *Proceedings of the 3rd innovations in theoretical computer science conference*. 2012, pp. 214–226.

[9] Sander Greenland, Judea Pearl, and James M Robins. "Confounding and collapsibility in causal inference". In: *Statistical science* 14.1 (1999), pp. 29–46.

[10] Will Hamilton, Zhitao Ying, and Jure Leskovec. "Inductive representation learning on large graphs". In: *Advances in neural information processing systems* 30 (2017).

[11] Moritz Hardt, Eric Price, and Nati Srebro. "Equality of opportunity in supervised learning". In: *Advances in neural information processing systems* 29 (2016).

[12] Fenyu Hu et al. "Graphair: Graph representation learning with neighborhood aggregation and interaction". In: *Pattern Recognition* 112 (2021), p. 107745.

[13] Zhimeng Jiang et al. "Fmp: Toward fair graph message passing against topology bias". In: *arXiv preprint arXiv:2202.04187* (2022).

[14] Yizhu Jiao et al. "Sub-graph contrast for scalable self-supervised graph representation learning". In: *2020 IEEE international conference on data mining (ICDM)*. IEEE. 2020, pp. 222–231.

[15] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[16] Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907* (2016).

[17] Matt J Kusner et al. "Counterfactual fairness". In: *Advances in neural information processing systems* 30 (2017).

[18] Tai Le Quy et al. "A survey on datasets for fairness-aware machine learning". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 12.3 (2022), e1452.

[19] Yuan Li et al. "EGNN: Constructing explainable graph neural networks via knowledge distillation". In: *Knowledge-Based Systems* 241 (2022), p. 108345.

[20] Christos Louizos et al. "Causal effect inference with deep latent-variable models". In: *Advances in neural information processing systems* 30 (2017).

[21] Sitao Luan et al. "Revisiting heterophily for graph neural networks". In: *Advances in neural information processing systems* 35 (2022), pp. 1362–1375.

[22] Jing Ma et al. "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.

[23] Yao Ma et al. "A unified view on graph neural networks as graph signal denoising". In: *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2021, pp. 1202–1211.

[24] David Madras et al. "Fairness through causal awareness: Learning causal latent-variable models for biased data". In: *Proceedings of the conference on fairness, accountability, and transparency*. 2019, pp. 349–358.

[25] Christopher Morris et al. "Weisfeiler and leman go neural: Higher-order graph neural networks". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 4602–4609.

[26] Alexandra Olteanu et al. "Social data: Biases, methodological pitfalls, and ethical boundaries". In: *Frontiers in big data* 2 (2019), p. 13.

[27] Judea Pearl. "Simpson's paradox, confounding, and collapibility". In: *Causality: models, reasoning and inference* (2009), pp. 173–200.

[28] Felix Petersen et al. "Post-processing for individual fairness". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 25944–25955.

[29] Tahleen Rahman et al. "Fairwalk: Towards fair graph embedding". In: (2019).

[30] Nripsuta Ani Saxena, Wenbin Zhang, and Cyrus Shahabi. "Missed opportunities in fair AI". In: *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM. 2023, pp. 961–964.

[31] Saeed Sharifi-Malvajerdi, Michael Kearns, and Aaron Roth. "Average individual fairness: Algorithms, generalization and experiments". In: *Advances in neural information processing systems* 32 (2019).

[32] Yongduo Sui et al. "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.

[33] Petar Veličković et al. “Graph attention networks”. In: *arXiv preprint arXiv:1710.10903* (2017).

[34] S Verma and J Rubin. “Fairness Definitions Explained. 2018 IEEE”. In: *ACM International Workshop on Software Fairness (FairWare), Gothenburg, Sweden.* 2018.

[35] Huaiyu Wan et al. “Aminer: Search and mining of academic social networks”. In: *Data Intelligence* 1.1 (2019), pp. 58–76.

[36] Tao Wang et al. “Powerful graph convolutional networks with adaptive propagation mechanism for homophily and heterophily”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 36. 4. 2022, pp. 4210–4218.

[37] Zichong Wang and Wenbin Zhang. “Group Fairness with Individual and Censorship Constraints”. In: *27th European Conference on Artificial Intelligence*. 2024.

[38] Zichong Wang et al. “: Fairness-Aware Graph Generative Adversarial Networks”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2023, pp. 259–275.

[39] Zichong Wang et al. “Advancing graph counterfactual fairness through fair representation learning”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2024, pp. 40–58.

[40] Zichong Wang et al. “FG-SMOTE: Towards Fair Node Classification with Graph Neural Network”. In: *ACM SIGKDD Explorations Newsletter, 2025* (2025).

[41] Zichong Wang et al. “FG^{2}AN: Fairness-aware Graph Generative Adversarial Networks”. In: *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*. Turin, Italy, 2023.

[42] Zichong Wang et al. “History, Development, and Principles of Large Language Models-An Introductory Survey”. In: *AI and Ethics, 2024* (2024).

[43] Zichong Wang et al. “Individual Fairness with Group Awareness Under Uncertainty”. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer Nature Switzerland. 2024, pp. 89–106.

[44] Zichong Wang et al. “Mitigating multisource biases in graph neural networks via real counterfactual samples”. In: *2023 IEEE International Conference on Data Mining (ICDM)*. IEEE. 2023, pp. 638–647.

[45] Zichong Wang et al. “Preventing Discriminatory Decision-making in Evolving Data Streams”. In: *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*. 2023.

[46] Zichong Wang et al. “Toward Fair Graph Neural Networks via Real Counterfactual Samples”. In: *Knowledge and Information Systems* (2024), pp. 1–25.

[47] Zichong Wang et al. “Towards Fair Graph Pooling with Group and Individual Awareness”. In: *proceedings of the AAAI conference on artificial intelligence*. 2024.

[48] Zichong Wang et al. “Towards fair machine learning software: Understanding and addressing model bias through counterfactual thinking”. In: *arXiv preprint arXiv:2302.08018* (2023).

[49] Jiancan Wu et al. “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.

[50] Zonghan Wu et al. “A comprehensive survey on graph neural networks”. In: *IEEE transactions on neural networks and learning systems* 32.1 (2020), pp. 4–24.

[51] Keyulu Xu et al. “How powerful are graph neural networks?” In: *arXiv preprint arXiv:1810.00826* (2018).

[52] I-Cheng Yeh and Che-hui Lien. “The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients”. In: *Expert systems with applications* 36.2 (2009), pp. 2473–2480.

[53] Zhipeng Yin, Zichong Wang, and Wenbin Zhang. “Improving Fairness in Machine Learning Software via Counterfactual Fairness Thinking”. In: *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings*. 2024, pp. 420–421.

[54] Si Zhang et al. “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.

[55] Wenbin Zhang. “AI fairness in practice: Paradigm, challenges, and prospects”. In: *Ai Magazine* (2024).

[56] Wenbin Zhang, Tina Hernandez-Boussard, and Jeremy Weiss. “Censored fairness through awareness”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 37. 12. 2023, pp. 14611–14619.

[57] Wenbin Zhang and Eirini Ntoutsi. “Faht: an adaptive fairness-aware decision tree classifier”. In: *arXiv preprint arXiv:1907.07237* (2019).

[58] Wenbin Zhang and Jeremy C Weiss. “Fairness with censorship and group constraints”. In: *Knowledge and Information Systems* 65.6 (2023), pp. 2571–2594.

[59] Wenbin Zhang and Jeremy C Weiss. “Longitudinal fairness with censorship”. In: *proceedings of the AAAI conference on artificial intelligence*. Vol. 36. 11. 2022, pp. 12235–12243.

[60] Wenbin Zhang et al. “Fairness amidst non-iid graph data: A literature review”. In: *arXiv preprint arXiv:2202.07170* 2 (2022).

[61] Wenbin Zhang et al. “Individual Fairness Guarantee in Learning with Censorship”. In: *arXiv preprint arXiv:2302.08015* (2023).

[62] Tong Zhao et al. “Learning from counterfactual links for link prediction”. In: *International Conference on Machine Learning*. PMLR. 2022, pp. 26911–26926.

[63] Yuchang Zhu et al. “FairAGG: Toward Fair Graph Neural Networks via Fair Aggregation”. In: *IEEE Transactions on Computational Social Systems* (2024).