Generating Diagnostic and Actionable Explanations for Fair Graph Neural Networks

Zhenzhong Wang[1], **Qingyuan Zeng**[2,3], **Wanyu Lin**[1*], **Min Jiang** [2,3], **Kay Chen Tan** [1]

[1] Department of Computing, The Hong Kong Polytechnic University

[2] School of Informatics, Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan, Ministry of Culture and Tourism, Xiamen University

[3] Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University

zhenzhong16.wang@connect.polyu.hk, 36920221153145@stu.xmu.edu.cn, wan-yu.lin@polyu.edu.hk, minjiang@xmu.edu.cn, kctan@polyu.edu.hk

Abstract

A plethora of fair graph neural networks (GNNs) have been proposed to promote algorithmic fairness for high-stake real-life contexts. Meanwhile, explainability is generally proposed to help machine learning practitioners debug models by providing human-understandable explanations. However, seldom work on explainability is made to generate explanations for fairness diagnosis in GNNs. From the explainability perspective, this paper explores the problem of *what subgraph patterns cause the biased behavior of GNNs, and what actions could practitioners take to rectify the bias?* By answering the two questions, this paper aims to produce compact, diagnostic, and actionable explanations that are responsible for discriminatory behavior. Specifically, we formulate the problem of generating diagnostic and actionable explanations as a multi-objective combinatorial optimization problem. To solve the problem, a dedicated multi-objective evolutionary algorithm is presented to ensure GNNs' explainability and fairness in one go. In particular, an influenced nodes-based gradient approximation is developed to boost the computation efficiency of the evolutionary algorithm. We provide a theoretical analysis to illustrate the effectiveness of the proposed framework. Extensive experiments have been conducted to demonstrate the superiority of the proposed method in terms of classification performance, fairness, and interpretability.

Introduction

To facilitate algorithmic fairness of machine learning in high-stakes applications, fair learning algorithms have recently attracted significant attention (Mehrabi et al. 2021; Wan et al. 2023; Jiang et al. 2022). Recently, several studies have been proposed towards fair graph neural networks (GNNs) due to their superior predictive performance on high-stakes applications (Jin et al. 2020; Wang et al. 2022a; Dai and Wang 2022; Wang et al. 2023a). On the other hand, *explainability*, a well-known principle to gain insights into the models, could help system developers and machine learning practitioners to debug the model to ensure algorithmic fairness by identifying data biases in training data (Ma et al. 2022a; Ge et al. 2022; Fu et al. 2020; Lin, Lan, and Li 2021). For example, Gopher (Pradhan et al. 2022) identifies the root of bias in the context of tabular data by identifying samples of biased patterns. BIND (Dong et al. 2023) quantifies the node's influences on the unfairness of GNNs based on the probabilistic distribution disparity. REFEREE (Dong et al. 2022) attempts to provide instance-level explanations for the bias of a certain node.

A primary issue of fair machine learning is that biases are usually introduced during the data collection. In particular, the topological structure of graph data might magnify the bias through the message-passing mechanisms (Dai and Wang 2022; Rouzrokh et al. 2022). However, existing explainable works are not specifically for model-level fairness diagnosis of GNNs to identify such topological structures and could not answer the following questions: *What subgraph patterns of the graph cause the biased behavior of GNNs, and what actions could machine learning practitioners take toward a fairer GNN?* To complement prior explainable approaches, this paper attempts to produce compact, diagnostic, and actionable subgraph patterns that are responsible for the model's biased behavior. Taking crime forecasting systems as an example, individuals within the same race often reside in dense communities, where edges denote social connections (shown in Fig. 1 (a)). Certain connection patterns may exacerbate algorithmic bias toward one specific race. Diagnostic explanations are desirable for machine learning practitioners to track the root of bias. Once the root of bias is identified, actionable explanations are provided for modifying the connection patterns, thus reducing the data bias and rectifying biased behaviors.

Intuitively, identifying a subset of edges and nodes from the entire graph to constitute subgraph patterns as explanations can be formulated as a combinatorial optimization problem. Nevertheless, it is challenging to solve the problem for the following three reasons: **1) Large-scale search space**: The combinations of edges and nodes for an explanation are exponential, and the explicit expressions of the gradients are difficult to obtain. It is, therefore, non-trivial to search from a large-scale space without the gradient information. **2) Multi-objective optimization**: Three objectives need to be

*Corresponding author.

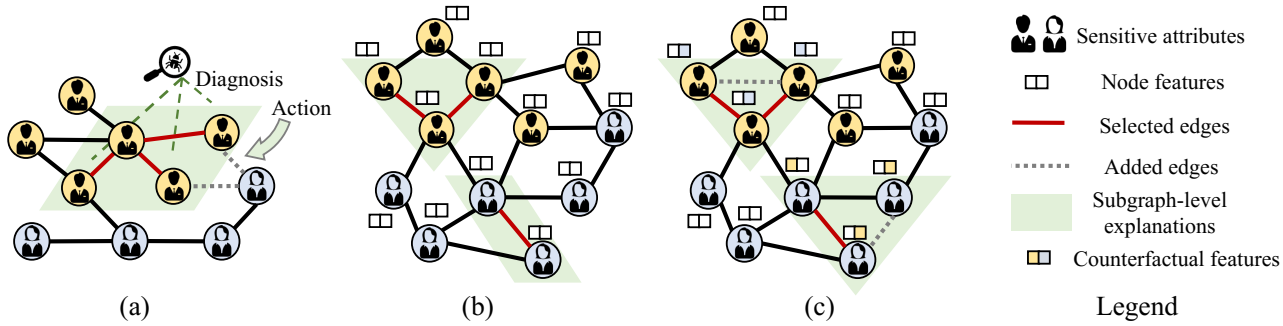

Figure 1: (a) An example of diagnostic explanations and actionable explanations on graphs. (b) Biased subgraphs as diagnostic explanations are identified by selecting a subset of edges and nodes to locate the root cause of the biased behavior of a GNN model. (c) By repairing the biased subgraphs (both features and structures), counterfactual debiased subgraphs as actionable explanations are generated to help the system developers and machine learning practitioners to reduce the data bias, thus rectifying discriminatory behavior.

optimized, including *Compactness*: the explanations should be compact enough so that humans can easily interpret them and take further actions (Lin et al. 2022); *Bias attribution*: The generated explanations should be able to capture the root of bias or rectify biased behaviors at large; *Accuracy*: Taking the generated explanations to modify the graph for fairness should not impair the propagation of useful information and thus does not affect model predictive performance. **3) Intensive computation**: Evaluating the three objective functions on the modified graph requires GNN retraining, and repetitively retraining the GNN towards a fair one is computation-intensive.

Inspired by the distinctive competency of evolutionary algorithms for large-scale combinatorial optimization problems (Zhou et al. 2021; Zhao et al. 2020; Wang et al. 2022b), we develop a dedicated multi-objective evolutionary algorithm-based framework for generating diagnostic and actionable explanations. The generated explanations can satisfy the following three objectives simultaneously: *compactness*, *bias attribution*, and *accuracy*. Moreover, we propose an influenced nodes-based gradient approximation to address the problem of expensive evaluation costs for updating the GNN model. Our proposed approximation algorithm could avoid retraining the GNN from scratch when evaluating the GNN on the modified graph. Extensive experiments on three benchmarking datasets are conducted to evaluate the generated diagnostic and actionable explanations. The experimental results demonstrate that the generated explanations can identify the root of bias and rectify biased behaviors. Specifically, we empirically show that the generated explanations can alleviate bias and reduce the statistical parity by $3\% \sim 9\%$ compared to vanilla GNNs while achieving comparable classification performance. In addition, we provide the theoretical fairness risk bound of the proposed method, indicating that using actionable explanations to rectify the root of bias can theoretically improve fairness.

Preliminaries and Problem Definition

In this section, we present the concepts of fair node classification and the related notions about diagnostic and actionable explanations. Several related works are briefly presented in Section Related Work (Appendix).

Notations: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ denotes an attributed graph with a set of nodes $\mathcal{V}$ and a set of edges $\mathcal{E}$, where $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_{|\mathcal{V}|})^\top \in \mathbb{R}^{|\mathcal{V}| \times d}$ is a set of node features. Each node i might have a sensitive attribute $s_i \in \{0, 1\}$. Taking binary classification as an example, fair node classification aims to ensure that the classifier $f : \mathbf{x} \rightarrow y$ makes fair predictions devoid of discrimination with respect to the sensitive attribute s, where $y \in \{0, 1\}$ denotes the node label.

Statistical parity and equal opportunity (Dwork et al. 2012; Hardt, Price, and Srebro 2016) are two widely used definitions for fairness.

Definition 1 *Statistical parity (SP) (Dwork et al. 2012) requires the classifier f to make identical prediction $\hat{y} \in \{0, 1\}$ while having different values of sensitive attribute s,* $\Pr(\hat{y}|s = 1) = \Pr(\hat{y}|s = 0)$.

Definition 2 *Equal opportunity (EO) (Hardt, Price, and Srebro 2016) requires the predictions of both groups with different values of the attribute to have the same true positive rate,* $\Pr(\hat{y}|y = 1, s = 1) = \Pr(\hat{y}|y = 1, s = 0)$.

Based on SP and EO, the fairness metrics can be defined as (Beutel et al. 2017; Louizos et al. 2015), $\Delta_{SP} = |\Pr(\hat{y}|s = 1) - \Pr(\hat{y}|s = 0)|$, $\Delta_{EO} = |\Pr(\hat{y}|y = 1, s = 1) - \Pr(\hat{y}|y = 1, s = 0)|$. The smaller the value of Δ_{SP} and Δ_{EO}, the fairer the model is.

Problem definition: We introduce a new formulation of explanation generation for fairness by formulating the problem as a multi-objective combinatorial optimization problem,

$$< \min \mathcal{C}, \max \mathcal{B}, \max \mathcal{F} >, \tag{1}$$

where three objective functions (*i.e., compactness $\mathcal{C}$, bias attribution $\mathcal{B}$, and accuracy $\mathcal{F}$*) are simultaneously optimized to generate subgraph patterns. Specifically, we aim to search biased subgraphs as diagnostic explanations $\mathcal{G}^{'} = \{\mathcal{G}_i^{'}\}$ for locating the bias of graph data (see Fig. 1 (b)). Furthermore, by modifying the biased subgraphs, we generate counterfactual debiased subgraphs as actionable explanations $\mathcal{G}^{''} = \{\mathcal{G}_i^{''}\}$

to help the system developers to mitigate the bias (see Fig. 1 (c)). Formally, we introduce the definitions of *compactness* and *bias attribution* of the generated subgraph patterns as shown below.

Definition 3 *(Compactness) Given a subgraph $\mathcal{G}_i^{(\cdot)}$ ($\mathcal{G}_i^{(\cdot)}$ denotes $\mathcal{G}_i^{'}$ or $\mathcal{G}_i^{''}$) with the node set $\mathcal{V}_i^{(\cdot)}$ and the edge set $\mathcal{E}_i^{(\cdot)}$, the compactness of the subgraph $\mathcal{G}_i^{(\cdot)}$ can be defined as,*

$$\mathcal{C}(\mathcal{G}_i^{(\cdot)}) = \frac{|\mathcal{E}_i^{(\cdot)}|}{|\mathcal{V}_i^{(\cdot)}|}. \tag{2}$$

The fewer edges included in the explanation, the easier it is for humans to interpret the subgraph (Lin et al. 2022).

Biased subgraph $\mathcal{G}_i^{'}$ is expected to be removed to suppress the propagation of biased information. Here, we use *bias attribution* to measure the improvement of fairness when subgraph $\mathcal{G}_i^{'}$ is removed from the raw graph.

Definition 4 *(Bias attribution of a biased subgraph) Given a subgraph $\mathcal{G}_i^{'}$, the bias attribution of $\mathcal{G}_i^{'}$ can be defined as,*

$$\mathcal{B}(\mathcal{G}_i^{'}) = \mathcal{M}(\mathcal{G};\theta) - \mathcal{M}(\mathcal{G}\backslash\mathcal{G}_i^{'};\theta^{'}), \tag{3}$$

where $\mathcal{M}(\mathcal{G}\backslash\mathcal{G}_i^{'};\theta^{'})$ means the fairness value (*e.g.*, Δ_{SP}) of the classifier $f_\theta^{'}$ trained on the modified graph $\mathcal{G}\backslash\mathcal{G}_i^{'}$. The larger the value of $\mathcal{B}(\mathcal{G}_i^{'})$, the greater the $\mathcal{G}_i^{'}$ toward bias.

By repairing the biased subgraph $\mathcal{G}_i^{'}$, counterfactual debiased subgraph $\mathcal{G}_i^{''}$ can be generated as actionable explanations. We extend the concept of bias attribution to the counterfactual debiased subgraph $\mathcal{G}_i^{''}$.

Definition 5 *(Bias attribution of a counterfactual debiased subgraph) Given a counterfactual debiased subgraph $\mathcal{G}_i^{''}$, the bias attribution of $\mathcal{G}_i^{''}$ can be defined as,*

$$\mathcal{B}(\mathcal{G}_i^{''}) = \mathcal{M}(\mathcal{G};\theta) - \mathcal{M}((\mathcal{G}\backslash\mathcal{G}_i^{'}) \cup \mathcal{G}_i^{''};\theta^{''}), \tag{4}$$

where $\theta^{''}$ is the classifier trained on modified graph $(\mathcal{G}\backslash\mathcal{G}_i^{'}) \cup \mathcal{G}_i^{''}$. In this work, both node features and structure can be revised in $\mathcal{G}_i^{'}$ to generate $\mathcal{G}_i^{''}$. The larger the value of $\mathcal{B}(\mathcal{G}_i^{''})$, the greater ability of $\mathcal{G}_i^{''}$ to reduce bias. We expect the counterfactual debiased subgraph should be as compact as possible so that the generated subgraphs are actionable for system developers to rectify the biased behaviors.

On top of bias attribution, we define the model-level (un)fairness explanations to guide the fairness debugging for system developers.

Definition 6 *(Model-level explanations for (un)fairness) Let f_θ denote a GNN classification model, our goal is to investigate what subgraph patterns $\mathcal{G}^{'}$ from the raw graph $\mathcal{G}$ account for the unfairness of the GNN model at large and what counterfactual subgraph patterns $\mathcal{G}^{''}$ can improve the fairness of the model. The obtained subgraph patterns $\mathcal{G}^{'}$ and $\mathcal{G}^{''}$ can be respectively treated as the attribution of the unfairness and fairness of the model,*

$$\begin{aligned}\mathcal{G}^{'} &= \arg\max_{\{\mathcal{G}_i^{'}\}} \mathcal{B}(\mathcal{G}_i^{'}), s.t.\ \mathcal{B}(\mathcal{G}_i^{'}) > 0 \\ \mathcal{G}^{''} &= \arg\max_{\{\mathcal{G}_i^{''}\}} \mathcal{B}(\mathcal{G}_i^{''}), s.t.\ \mathcal{B}(\mathcal{G}_i^{''}) > 0.\end{aligned} \tag{5}$$

We formulate the model-level explanation generation problem as Eq.(1) to maintain the explainability, fairness, and classification performance simultaneously.

Proposed Framework

The proposed framework for *G*enerating *e*xplanations for *b*ias (*Geb*) is illustrated in Fig. 2. The framework includes two stages: identification of biased subgraphs for locating the bias patterns (top branch), and generation of counterfactual debiased subgraphs for mitigating the bias (bottom branch).

Identification of Biased Subgraphs

Due to the remarkable competency of evolutionary algorithms in addressing combinatorial optimization problems and other complex optimization problems (Wang et al. 2023b; Jiang et al. 2021b; He et al. 2022; Jiang et al. 2021c; Wang et al. 2024; Jiang et al. 2021a), this work developed a multi-objective evolutionary algorithm to solve the explanation generation problem. Evolutionary algorithms search from a set of initial solutions through genetic operators to create new sets of better solutions to search toward the optima. Firstly, a set of subgraphs is randomly sampled from the graph data as the initial population. By performing the genetic operators, *i.e.,* crossover and mutation, to the population, a set of newly generated subgraphs is produced. Then, three objective functions, *i.e.*, the compactness and bias attribution of each subgraph as well as the model accuracy, are evaluated. After that, the subgraphs with better objective values are selected as the parent population for the next generation. The above processes are repeatedly conducted until the terminal condition (*e.g.*, the maximum number of generations) is met, and the subgraphs of the last generation are kept as the diagnostic explanations. The major components of our multi-objective evolutionary algorithm are elaborated below:

Population: The population refers to a set of candidate subgraphs $\mathcal{G}^{'} = \{\mathcal{G}_i^{'}\}$, where $i = 1, ..., I$, and I is the population size. In this work, we randomly sample I nodes and their 1-hop neighbors as the initial subgraphs.

Crossover: Crossover operators play a critical role in generating new solutions with better convergence by mixing the attributes of two solutions, which enables the newly produced solutions to have beneficial combinations of attributes of their parent. In this work, two candidate subgraphs are randomly selected as the parent subgraphs from the population pool to perform the crossover operator so that an offspring subgraph $\mathcal{G}_{of}^{'}$ can be generated. The crossover operator mixes the edges of the two selected subgraphs $\mathcal{G}_1^{'} = \{\mathcal{V}_1^{'}, \mathcal{E}_1^{'}\}$ and $\mathcal{G}_2^{'} = \{\mathcal{V}_2^{'}, \mathcal{E}_2^{'}\}$,

$$\mathcal{G}_{of}^{'} = (\mathcal{V}_{of}^{'}, (\mathcal{E}_1^{'} \cap \mathcal{E}_2^{'}) \cup \Pr(\mathcal{E}_1^{'}\backslash\mathcal{E}_2^{'}) \cup \Pr(\mathcal{E}_2^{'}\backslash\mathcal{E}_1^{'})), \tag{6}$$

where $\mathcal{V}_{of}^{'}$ is the set of nodes that are connected by the edges in $\mathcal{G}_{of}^{'}$, and $\Pr(\cdot)$ randomly outputs a subset of edges from the candidate edges. As such, the common edges in both parents that may amplify bias will be kept in offspring, and potentially biased edges of one parent are randomly kept.

Mutation: By modifying the partial attributes in solutions, mutation operators introduce randomness and variability into

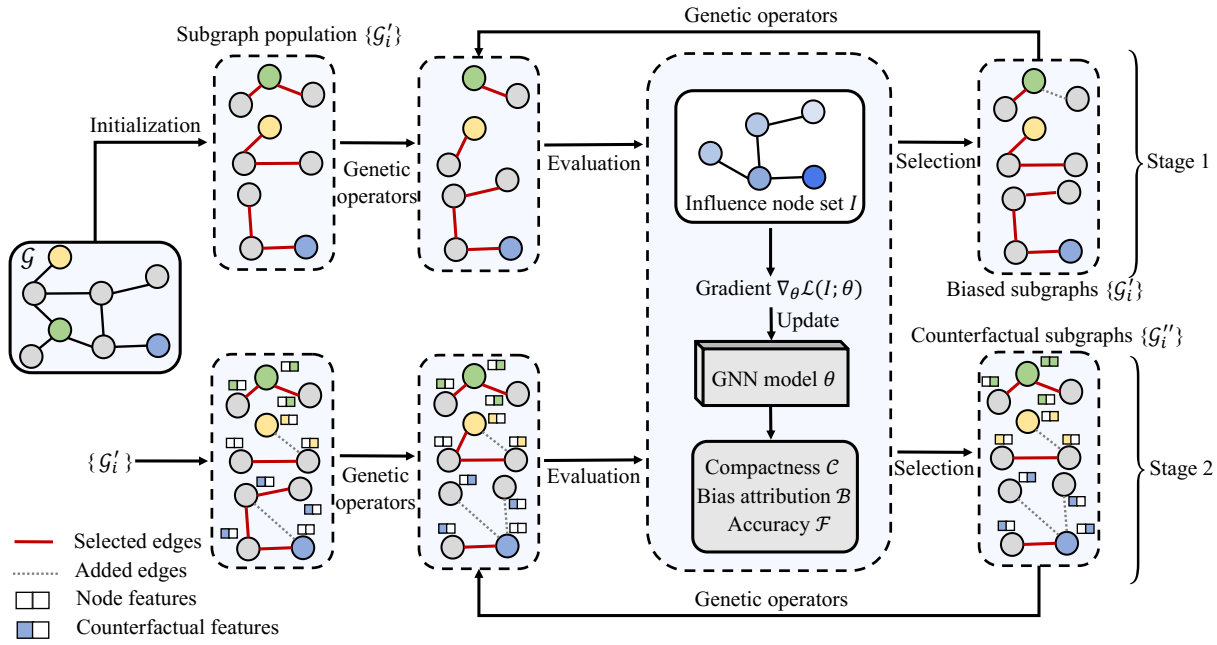

Figure 2: Two stages in the proposed *Geb*: The top branch (stage one) aims to identify biased subgraphs as diagnostic explanations for locating the root of data bias. The bottom branch (stage two) aims to generate counterfactual debiased subgraphs as actionable explanations for mitigating the bias.

the population during the evolution process, which can enhance the diversity of the population and help the population to escape from the local optima. In our design, for each candidate subgraph, we randomly add one edge that connects the subgraph to its adjacent nodes or randomly drop one edge from the subgraph to generate a new subgraph.

Objective Function Evaluation: Each subgraph in the population pool will get objective values that measure the quality of the generated biased subgraph. Specifically, the compactness $\mathcal{C}(\mathcal{G}_i^{'})$ and the bias attribution $\mathcal{B}(\mathcal{G}_i^{'})$ of a subgraph $\mathcal{G}_i^{'}$ are evaluated by Eq. (2) and Eq. (3), respectively. The model accuracy $\mathcal{F}$ is the AUC score obtained by the GNN trained on the modified graph $\mathcal{G}\backslash\mathcal{G}_i^{'}$.

Selection: Given the objective values of each subgraph, we use the non-dominated sort to select the non-dominated solutions as the parent population for the next generation (Tan, Khor, and Lee 2005). The definition of Pareto dominance is provided in Section Multiobjective Optimization (Appendix).

Influenced Nodes-based Gradient Approximation

It is extremely computationally expensive to retrain the GNN model to evaluate the objective functions on the modified graph. To address the computational challenge, we assume that modifying the local structure of a graph may not have a great impact on the distant nodes, and the GNN model can be simply updated on the influenced nodes affected by $\mathcal{G}^{'}$. The rationality comes from the widely used local-dependence assumption for graph data (Wu et al. 2020). Based on this assumption, we proposed an influenced nodes-based gradient approximation to update the GNN model with a lower computational cost. First, we give the definition of the influenced nodes,

Definition 7 *(Influenced nodes) Assume the change of node u's embedding is $\Delta\mathbf{h}_u$ after modifying the graph data, the set of influenced nodes can be defined as,*

$$\mathcal{I} = \{u | \arg\max_i \sigma(\mathbf{h}_u)[i] \neq \arg\max_i \sigma(\mathbf{h}_u + \Delta\mathbf{h}_u)[i]\}, \tag{7}$$

where $u \in \Omega_l(\mathcal{G}^{'})$, $\Omega_l(\mathcal{G}^{'})$ is the node set of l-hop neighbors of all nodes in $\mathcal{G}^{'}$, l is the layer number of the studied GNN, σ is the softmax function, $\mathbf{h}_u$ is the embedding of the node u before modfiying the graph, and $\sigma(\mathbf{h}_u)[i]$ is the predicted probability of the node u belonging to the class i. The definition indicates that those nodes from $\Omega_l(\mathcal{G}^{'})$ whose classification results are different after modifying the graph are the influenced nodes. We update the GNN model on those influenced nodes instead of the whole graph to save computational costs.

Another issue is estimating the change of node u's embedding, $\Delta\mathbf{h}_u$. We design a fast approximation algorithm to estimate the change of a node's embedding. For each node $u \in \Omega_l(\mathcal{G}^{'})$, the change $\Delta\mathbf{h}_u$ depends on the l-hop neighbors $\Omega_l(u)$ and the l-hop removed neighbors $\overline{\Omega}_l(u)$.

$$\Delta\mathbf{h}_u = \sum_{v \in \Omega_l(u)} s_v^l \mathbf{h}_v - \sum_{v \in \overline{\Omega}_l(u)} s_v^l \mathbf{h}_v. \tag{8}$$

For each node u, s_u^l can be calculated as,

$$s_u^l = \frac{\sum_{v \in (\Omega_l(u) \cup \overline{\Omega}_l(u) \cup u)} s_v^{l-1}}{d_u + 1}, s_u^0 = 1, s_v^0 = 0, v \neq u, \tag{9}$$

where d_u is the degree of the node u. The intuition behind the influence scores is that the greater the distance between a node v to u, the smaller the influence score s_v of $\mathbf{h}_v$, and the less information is propagated from $\mathbf{h}_v$ for changing $\mathbf{h}_u$. For a better understanding, an illustrative example of calculating the influence score is provided in Section Evaluation of Influence Scores (Appendix).

After estimating $\Delta\mathbf{h}$, the influenced nodes $\mathcal{I}$ can be identified by Eq. (7). Assuming that model parameters do not change significantly when a small subset of data points is changed, the model parameters can get a minimal change (Jagielski et al. 2021), thus the parameters of the GNN model can be updated on the set of influenced nodes by a single step of gradient descent,

$$\theta^{'} = \theta - \eta \left(\nabla_\theta \mathcal{L}(\mathcal{G}, \theta) - \frac{1}{|\mathcal{I}|} \nabla_\theta \mathcal{L}(\mathcal{I}, \theta) \right), \tag{10}$$

where η is the learning rate for the gradient step and θ is the model trained on the raw graph $\mathcal{G}$. In this way, based on the trained model θ, we update the GNN to evaluate the objective functions instead of retraining the GNN from scratch.

Generation of Counterfactual Debiased Subgraphs

After identifying the biased subgraphs, we further seek to provide counterfactual debiased subgraphs as actionable explanations for system developers to repair such roots of bias. Taking the biased subgraphs $\mathcal{G}^{'}$ as the initial subgraphs, the counterfactual debiased subgraphs $\mathcal{G}^{''}$ are also generated by the proposed multi-objective evolutionary algorithm. Due to the counterfactual nature, different from stage one, the mutation operator in stage two can add previously non-existent edges in the raw graph to generate counterfactual structures. The model accuracy is the AUC score obtained by the classifier trained on $(\mathcal{G} \backslash \mathcal{G}_i^{'}) \cup \mathcal{G}_i^{''}$. Other components in the multi-objective evolutionary algorithm are the same as those of stage one.

In addition to generating counterfactual structures, we also seek to generate counterfactual features. Given a node $v \in \mathcal{G}_i^{''}$, the counterfactual features $\mathbf{x}_v^c$ can be obtained by minimizing the fairness constraint $\mathcal{L}_{cov}$ with the gradient descent algorithm,

$$\mathbf{x}_v^c = \Pi(\mathbf{x}_v - \eta \nabla_{\mathbf{x}_v^c} \mathcal{L}_{cov}), \tag{11}$$

where Π projects the generated features back to the feasible domain, and $\mathcal{L}_{cov}$ is the absolute covariance between the sensitive attribute s and the prediction $\hat{y}$ (Dai and Wang 2021),

$$\mathcal{L}_{cov} = |\mathbb{E}[(s - \mathbb{E}(s)(\hat{y} - \mathbb{E}(\hat{y}))]|. \tag{12}$$

The computation of the gradient $\nabla_{\mathbf{x}_v^c} \mathcal{L}_{cov}$ is complicated. Here, we derive the fairness constraint with respect to $\mathbf{x}_v^c$ via the chain rule,

$$\frac{\partial \mathcal{L}_{cov}}{\partial \mathbf{x}_v^c} = \frac{\partial \mathcal{L}_{cov}}{\partial \theta} \frac{\partial \theta}{\partial \mathbf{x}_v^c}, \tag{13}$$

where $\frac{\partial \mathcal{L}_{cov}}{\partial \theta}$ is the gradient w.r.t $\mathcal{L}_{cov}$, and we denote it as $\nabla_\theta \mathcal{L}_{cov}$. Then, $\frac{\partial \theta}{\mathbf{x}_v^c}$ can be obtained by (Koh and Liang 2017),

$$\frac{\partial \theta}{\mathbf{x}_v^c} = -H_\theta^{-1} \frac{\partial^2 \mathcal{L}_{cov}}{\partial \theta \partial \mathbf{x}_v^c}, \tag{14}$$

where H_θ is the Hessian of the loss. By combining Eq. (13) and Eq. (14), the gradient of the loss w.r.t. the generated counterfactual features $\mathbf{x}_v^c$ is,

$$\frac{\partial \mathcal{L}_{cov}}{\partial \mathbf{x}_v^c} = -(\nabla_\theta \mathcal{L}_{cov})^\top H_\theta^{-1} \frac{\partial^2 \mathcal{L}_{cov}}{\partial \theta \partial \mathbf{x}_v^c}. \tag{15}$$

After generating offspring subgraphs with counterfactual structures and features, the evolutionary process is iteratively performed until the terminal condition is satisfied. The subgraphs of the last generation are served as the actionable explanations for diluting the bias. The pseudo-code of the two stages can be seen in Section Our Algorithm: *Geb* (Appendix).

Theoretical Analysis

In this section, we provide the fairness risk bound to illustrate the effectiveness of the generated counterfactual debiased subgraphs in improving fairness. Firstly, The expectation fairness risk Δ_{SP} and empirical fairness risk $\tilde{\Delta}_{SP}$ of a classifier f are introduced,

$$\begin{aligned} &\Delta_{SP}(f, \mathcal{G}) = \\ &\qquad |\mathbb{E}_{S=s^+}[f_{\mathbf{x}\sim\mathcal{G}}(\mathbf{x}) = 1] - \mathbb{E}_{S=s^-}[f_{\mathbf{x}\sim\mathcal{G}}(\mathbf{x}) = 1]|, \\ &\tilde{\Delta}_{SP}(f, \mathcal{G}) = \\ &\qquad |\mathbb{I}_{S=s^+}(f_{\mathbf{x}\sim\mathcal{G}}(\mathbf{x}) = 1) - \mathbb{I}_{S=s^-}(f_{\mathbf{x}\sim\mathcal{G}}(\mathbf{x}) = 1)|, \end{aligned} \tag{16}$$

where $\mathcal{G}$ is a graph coming from the domain $\mathcal{Z}$, and $\mathbb{I}$ is an indicator function.

Theorem 1 *Assume that $\mathcal{F}$ is a function class consisting of functions with range $[a, b]$, for any classifier $f \in \mathcal{F}$ with a graph $\mathcal{G}$, if the difference between the empirical fairness risk $\tilde{\Delta}_{SP}$ and the expectation fairness risk Δ_{SP} can be bounded by $\tau_{\Delta_{SP}}$. Then, for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, the expectation fairness risk bound Δ_{SP} is given by,*

$$\begin{aligned} \Delta_{SP}(f, \mathcal{G}) \leq & \tilde{\Delta}_{SP}(f, \mathcal{G}^{(ce)}) + D_{\mathcal{F}}(\mathcal{Z}^{(ub)}, \mathcal{Z}^{(bi)}) \\ & + \tau_{\Delta_{SP}} + \sqrt{\frac{(b-a)^2 \ln(4/\delta)}{4n}}, \end{aligned} \tag{17}$$

where $\mathcal{G}$ is the raw graph from the domain $\mathcal{Z}^{(bi)}$ with inherent data bias, $\mathcal{G}^{(ce)}$ is the debiasing graph, $\mathcal{Z}^{(ub)}$ is the domain of ideal debiased data, and n is the number of nodes.

It should be noted that the model f in $\Delta_{SP}(f, \mathcal{G})$ and $\tilde{\Delta}_{SP}(f, \mathcal{G}^{(ce)})$ are not exactly the same, as the parameters of f have been updated on $\mathcal{G}^{(ce)}$. While according to Eq. (10), they can be approximately equivalent. The theoretical analysis shows that the expectation fairness risk $\Delta_{SP}(f, \mathcal{G})$ is bounded by the empirical fairness risk $\tilde{\Delta}_{SP}(f, \mathcal{G}^{(ce)})$, the difference $D_{\mathcal{F}}(\mathcal{Z}^{(ub)}, \mathcal{Z}^{(bi)})$ between the two domains $\mathcal{Z}^{(ub)}$ and $\mathcal{Z}^{(bi)}$, and two additional terms $\sqrt{\frac{(b-a)^2 \ln(4/\delta)}{4n}}$ and $\tau_{\Delta_{SP}}$, suggesting that generating counterfactual debiased subgraphs and reducing the inherent bias degree of the collected data (i.e., $D_{\mathcal{F}}(\mathcal{Z}^{(ub)}, \mathcal{Z}^{(bi)})$) can reduce the upper bound of the fairness risk. This work uses the generated

counterfactual debiased subgraphs to produce a debiasing graph $\mathcal{G}^{(ce)}$ with a fair classifier f to minimize the empirical fairness risk $\tilde{\Delta}_{SP}(f, \mathcal{G}^{(ce)})$, thus bounding the expectation fairness risk $\Delta_{SP}(f, \mathcal{G})$.The proof is given in Section The Fairness Risk Bound (Appendix).

Experiments

Experimental Settings

We evaluate *Geb* on three real-world graph datasets: NBA (Dai and Wang 2021), Pokec-z (Takac and Zabovsky 2012), and Pokec-n (Takac and Zabovsky 2012). The details of the datasets can be found in Section Experimental Setup (Appendix). To investigate the flexibility of the proposed *Geb*, we incorporate it into GCN (Kipf and Welling 2017), GAT (Veličković et al. 2017), NIFTY (Agarwal, Lakkaraju, and Zitnik 2021), FairGCN (Dai and Wang 2021), and FairGAT (Dai and Wang 2021), and rename them as *Geb*-GCN, *Geb*-GAT, *Geb*-NIFTY, *Geb*-FairGCN, and *Geb*-FairGAT, respectively. We compare the proposed *Geb* with several state-of-the-art methods: ALFR and ALFR-e (Edwards and Storkey 2015), Debias and Debias-e (Zhang, Lemoine, and Mitchell 2018), FCGE (Bose and Hamilton 2019), FairGCN (Dai and Wang 2021), FairGAT (Dai and Wang 2021), NT-FairGNN (Dai and Wang 2022), GE-REFEREE (Dong et al. 2022), GEAR (Ma et al. 2022b), and NIFTY (Agarwal, Lakkaraju, and Zitnik 2021). We use the AUC metric to measure the node classification performance. Δ_{SP} and Δ_{EO} are used to measure the fairness. The details of the experimental setup can be seen in Section Experimental Setup (Appendix).

Comparisons with Baselines

Table 1 shows the performance comparison of the compared algorithms. The best performance is highlighted in bold. Experimental results show that *Geb*-GCN, *Geb*-GAT, *Geb*-NIFTY, *Geb*-FairGCN, and *Geb*-FairGAT outperform their vanilla counterparts in terms of the fairness metrics Δ_{SP} and Δ_{EO}, *e.g.,* compared to vanilla counterparts, the proposed *Geb* reduces Δ_{SP} by $3\% \sim 9\%$ without losing too much classification performance.

We also plot the Pareto front obtained by FairGCN, NT-FairGNN, NIFTY-GCN, GEAR, *Geb*-GCN, and *Geb*-FairGCN, as shown in Fig. 10 (Appendix). It should be noted that our proposed *Geb* can provide a set of Pareto solutions in one run. As for the compared algorithms, we ran multiple times to obtain the Pareto solutions. Obviously, the proposed *Geb* often arrive at the lower-right corner, *i.e.,* high accuracy, low Δ_{SP}. That is, the solutions obtained by *Geb* Pareto dominate most of the solutions obtained by the compared algorithms. Moreover, we visualize of fairness performance of compared algorithms in terms of Δ_{SP} and Δ_{EO} in Fig. 4 (Appendix). Together with these experimental results, the effectiveness of applying the generated actionable explanations in enhancing fairness and maintaining classification performance can be demonstrated.

Interpretability Comparisons. To investigate the interpretability of the generated actionable explanations, we measure the compactness of generated explanations, as commonly done in the literature. However, seldom works have been specifically proposed to generate subgraph-level explanations for fairness debugging so far. As NIFTY and GEAR can alleviate bias by generating counterfactual graphs, we take the counterfactual graphs as the explanations for fairness. The compactness of explanations generated by *Geb*, GEAR, and NIFTY are presented in Table 1 (Appendix). It can be observed that *Geb* produces more compact explanations compared to GEAR and NIFTY. This is because NIFTY globally perturbs the structure to generate counterfactuals, making the explanations non-compact for humans to interpret. GEAR only perturbs features to output an entire graph instead of a set of compact subgraphs.

Moreover, two diagnostic explanations and two actionable explanations generated by *Geb* on the NBA dataset are plotted in Fig. 5 (a) (Appendix) and Fig. 5 (c) (Appendix), and Fig. 5 (b) (Appendix) and Fig. 5 (d) (Appendix), respectively. We can observe that the diagnostic explanations tend to remove positive edges (i.e., the edge that connects with the same sensitive attribute) to suppress the magnification of bias. Further, the actionable explanations are likely to add negative edges (i.e., the edge that connects with different sensitive attributes) to dilute bias. The explanation visualization aligns with human intuition, *i.e.,* nodes with the same sensitive attributes tend to be interconnected and could amplify biases.

In addition to visualizing explanatory subgraphs, we quantitatively analyze four types of edges (i.e., added positive edges, added negative edges, removed positive edges, and removed negative edges) to illustrate the behavior of *Geb* in reducing bias. From Fig. 6 (Appendix), we confirm that *Geb* tends to add negative edges and remove positive edges. Furthermore, inspired by this finding, we randomly flip positive edges to negative edges to see whether fairness can be improved while maintaining accuracy. The fairness evaluation and classification results are shown in Fig. 7 (Appendix). Notably, randomly flipping the positive edges may not necessarily improve fairness and maintain the classification performance, illustrating the rationality of using evolutionary algorithms to heuristically search biased patterns.

Ablation Studies

To verify the effectiveness of the generated biased subgraphs, we analyzed the performance of *Geb* at stage one (*Geb*-GCN-S1 and *Geb*-GAT-S1), *i.e.*, directly removing the biased subgraphs. It should be noted that only the edges of the biased subgraphs are removed from the graph data, but the nodes are preserved. From Table 2 (Appendix), we can see if we only remove the biased subgraph from the graph without performing stage two to repair them, the bias can also be reduced to some extent. This phenomenon shows that there are certain subgraph patterns in the graph that can magnify the bias, and the biased structures can be located by stage one of the proposed *Geb*. However, compared to *Geb*-GCN and *Geb*-GAT, directly removing the biased subgraphs can result in more loss of useful topology information, thus degrading the AUC scores more.

To validate the effectiveness of the proposed influenced nodes-based gradient approximation, we evaluate the perfor-

Method	AUC (↑)			Δ_{SP} (↓)			Δ_{EO} (↓)		
	NBA	Pokec-z	Pokec-n	NBA	Pokec-z	Pokec-n	NBA	Pokec-z	Pokec-n
GCN	**78.3±0.3**	**77.2±0.1**	**75.1±0.2**	7.9±1.3	9.9±1.1	9.6±0.9	17.8±2.6	9.1±0.6	12.8±1.3
GAT	78.2±0.6	76.7±0.1	75.1±0.2	10.2±2.5	9.1±0.9	9.4±0.7	15.9±4.0	8.4±0.6	12.0±1.5
ALFR	71.5±0.3	71.3±0.3	67.7±0.5	2.3±0.9	2.8±0.5	3.1±0.5	3.2±1.5	1.1±0.4	3.9±0.6
ALFR-e	72.9±1.0	74.0±0.7	71.9±0.3	4.7±1.8	5.8±0.4	4.1±0.5	4.7±1.7	2.8±0.8	4.6±1.6
Debias	71.3±0.7	71.4±0.6	67.9±0.7	2.5±1.5	1.9±0.6	2.4±0.7	3.1±1.9	1.9±0.4	2.6±1.0
Debias-e	72.9±1.2	74.2±0.7	71.7±0.7	5.3±0.9	4.7±1.0	3.6±0.2	3.1±1.3	3.0±1.4	4.4±1.2
FCGE	73.6±1.5	71.0±0.2	69.5±0.4	2.9±1.0	3.1±0.5	4.1±0.8	3.0±1.2	1.7±0.6	5.5±0.9
FairGCN	77.0±0.3	76.7±0.2	74.9±0.4	1.0±0.5	0.9±0.5	0.8±0.2	1.2±0.4	1.7±0.2	1.1±0.5
FairGAT	77.5±0.7	76.5±0.2	74.9±0.4	0.7±0.5	0.5±0.3	0.6±0.3	0.7±0.3	0.8±0.3	0.8±0.2
NT-FairGNN	77.0±0.3	76.7±0.3	74.9±0.4	1.0±0.5	1.0±0.4	0.8±0.2	1.2±0.4	1.6±0.2	1.1±0.3
GE-REFEREE	75.6±0.9	OOM	OOM	2.8±0.3	OOM	OOM	0.8±0.3	OOM	OOM
GEAR	70.0±1.0	70.1±1.5	69.1±1.1	2.0±0.5	2.2±0.2	2.0±0.7	0.9±0.3	3.4±0.3	6.0±0.4
NIFTY	67.7±0.8	70.9±1.1	68.5±1.5	3.0±0.8	0.8±0.5	4.7±0.5	5.8±0.9	4.0±1.3	6.7±0.7
Geb-NIFTY (Ours)	71.3±0.5	72.8±1.0	70.9±1.1	0.6±0.4	0.5±0.3	0.9±0.2	1.3±0.5	0.8±0.4	1.2±0.5
Geb-GCN (Ours)	74.6±0.8	71.4±0.2	72.6±0.4	0.5±0.4	**0.3±0.2**	6.3±0.4	0.6±0.2	**0.4±0.2**	4.3±0.6
Geb-GAT (Ours)	74.8±0.6	73.1±0.2	72.9±0.6	**0.4±0.3**	**0.3±0.2**	6.3±0.8	0.6±0.2	**0.4±0.1**	5.1±0.3
Geb-FairGCN (Ours)	74.9±0.6	72.3±0.2	72.4±0.2	**0.4±0.3**	0.4±0.3	**0.2±0.1**	**0.5±0.1**	**0.4±0.2**	0.3±0.1
Geb-FairGAT (Ours)	74.9±0.7	71.0±0.7	74.8±0.2	0.5±0.3	0.6±0.3	**0.2±0.1**	0.6±0.1	0.6±0.3	**0.2±0.1**

Table 1: The comparisons of our proposed *Geb* with the baselines (%). OOM: Out of memory.

mance of *Geb* w/o Influ (i.e., based on the latest parameters, update the model with a few steps of gradient descent) and *Geb*-FS (i.e., train the model from scratch). From Table 2 (Appendix), it can be found that *Geb* w/o Influ and *Geb*-FS can achieve performance that is not significantly different from the original ones. Besides, we evaluate the computational cost of *Geb*-GCN, and *Geb*-GAT, *Geb*-GCN-FS, *Geb*-GCN w/o Influ, *Geb*-GAT w/o Influ, and several baseline algorithms. As shown in Table 3 (Appendix), we can confirm that the proposed influenced nodes-based gradient approximation can cut running time to $10\% \sim 93\%$ without significantly affecting classification performance and fairness. Though *Geb* consumes more running time than other baselines, we argue that it is a one-time cost. We want to highlight that *Geb* can achieve competitive results in terms of fairness and AUC scores, and provide explainability — an essential aspect that other baselines do not have. To examine the impact of the generated counterfactual features, we evaluate the performance of *Geb* without the counterfactual features, and the corresponding ablation algorithms are renamed as *Geb* w/o cf. From Table 2 (Appendix), we can see that counterfactual features can further enhance fairness.

Sensitivity Analysis

To investigate the impact of genetic operators, we plot the AUC scores and the Δ_{SP} values under varying crossover rates and mutation rates in Fig. 8 (Appendix) and Fig. 9 (Appendix), respectively. Generally, a too-high mutation rate with a too-low crossover rate will result in poor search performance. The analysis of convergence and the population size can be found in Section Convergence Analysis (Appendix) and Population Size Analysis (Appendix), respectively.

Conclusion

In this work, we advanced a model-agnostic fair GNN framework to generate compact and interpretable model-level explanations for locating the source of bias and rectifying the biased behavior. In particular, a multi-objective evolutionary algorithm is designed to generate diagnostic explanations and actionable explanations. The former explains the root cause of the biased behavior of the GNNs, highlighting nodes or edges causing the bias. The latter provides counterfactuals on how to rectify this bias, demonstrating the specific nodes or edges that should be added or removed. To address the computational challenge in the evolutionary algorithm, we proposed an influenced nodes-based gradient approximation to update the GNN models. We also provided a theoretical fairness risk bound to illustrate the effectiveness of *Geb* in reducing the bias. The experimental results exhibit promising advancements in terms of classification performance, fairness, and interpretability.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant U21A20512 and Grant 62276222; in part by the Research Grants Council of the Hong Kong SAR under Grant PolyU11211521, Grant PolyU15218622, and Grant PolyU15215623; in part by The Hong Kong Polytechnic University (Project IDs: P0039734 and P0035379); in part by the PolyU Internal Research Fund under Grant P0042687; in part by the PolyU Start-up Fund under Grant P0046682; and in part by the General Research Fund of Hong Kong under Grant P0041813.

References

Agarwal, C.; Lakkaraju, H.; and Zitnik, M. 2021. Towards a unified framework for fair and stable graph representation

learning. In *Uncertainty in Artificial Intelligence*, 2114–2124. PMLR.

Beutel, A.; Chen, J.; Zhao, Z.; and Chi, E. H. 2017. Data decisions and theoretical implications when adversarially learning fair representations. *2017 Workshop on Fairness, Accountability, and Transparency in Machine Learning*.

Bose, A.; and Hamilton, W. 2019. Compositional fairness constraints for graph embeddings. In *International Conference on Machine Learning*, 715–724. PMLR.

Dai, E.; and Wang, S. 2021. 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*, 680–688.

Dai, E.; and Wang, S. 2022. Learning fair graph neural networks with limited and private sensitive attribute information. *IEEE Transactions on Knowledge and Data Engineering*.

Dong, Y.; Wang, S.; Ma, J.; Liu, N.; and Li, J. 2023. Interpreting Unfairness in Graph Neural Networks via Training Node Attribution. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Dong, Y.; Wang, S.; Wang, Y.; Derr, T.; and Li, J. 2022. On Structural Explanation of Bias in Graph Neural Networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, 316–326. New York, NY, USA: Association for Computing Machinery.

Dwork, C.; Hardt, M.; Pitassi, T.; Reingold, O.; and Zemel, R. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 214–226.

Edwards, H.; and Storkey, A. 2015. Censoring representations with an adversary. *The International Conference on Learning Representations*.

Fu, Z.; Xian, Y.; Gao, R.; Zhao, J.; Huang, Q.; Ge, Y.; Xu, S.; Geng, S.; Shah, C.; Zhang, Y.; et al. 2020. Fairness-aware explainable recommendation over knowledge graphs. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 69–78.

Ge, Y.; Tan, J.; Zhu, Y.; Xia, Y.; Luo, J.; Liu, S.; Fu, Z.; Geng, S.; Li, Z.; and Zhang, Y. 2022. Explainable fairness in recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 681–691.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29.

He, X.; Zheng, Z.; Chen, Z.; and Zhou, Y. 2022. Adaptive Evolution Strategies for Stochastic Zeroth-Order Optimization. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(5): 1271–1285.

Jagielski, M.; Severi, G.; Pousette Harger, N.; and Oprea, A. 2021. Subpopulation Data Poisoning Attacks. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21, 3104–3122. New York, NY, USA: Association for Computing Machinery.

Jiang, M.; Wang, Z.; Guo, S.; Gao, X.; and Tan, K. C. 2021a. Individual-Based Transfer Learning for Dynamic Multiobjective Optimization. *IEEE Transactions on Cybernetics*, 51(10): 4968–4981.

Jiang, M.; Wang, Z.; Hong, H.; and Yen, G. G. 2021b. Knee Point-Based Imbalanced Transfer Learning for Dynamic Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 25(1): 117–129.

Jiang, M.; Wang, Z.; Qiu, L.; Guo, S.; Gao, X.; and Tan, K. C. 2021c. A Fast Dynamic Evolutionary Multiobjective Algorithm via Manifold Transfer Learning. *IEEE Transactions on Cybernetics*, 51(7): 3417–3428.

Jiang, Z.; Han, X.; Fan, C.; Yang, F.; Mostafavi, A.; and Hu, X. 2022. Generalized Demographic Parity for Group Fairness. In *International Conference on Learning Representations*.

Jin, G.; Wang, Q.; Zhu, C.; Feng, Y.; Huang, J.; and Zhou, J. 2020. Addressing crime situation forecasting task with temporal graph convolutional neural network approach. In *2020 12th International Conference on Measuring Technology and Mechatronics Automation*, 474–478. IEEE.

Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. *International Conference on Learning Representations*.

Koh, P. W.; and Liang, P. 2017. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*, 1885–1894. PMLR.

Lin, W.; Lan, H.; and Li, B. 2021. Generative causal explanations for graph neural networks. In *International Conference on Machine Learning*, 6666–6679. PMLR.

Lin, W.; Lan, H.; Wang, H.; and Li, B. 2022. Orphicx: A causality-inspired latent variable model for interpreting graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13729–13738.

Louizos, C.; Swersky, K.; Li, Y.; Welling, M.; and Zemel, R. 2015. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*.

Ma, J.; Guo, R.; Mishra, S.; Zhang, A.; and Li, J. 2022a. CLEAR: Generative Counterfactual Explanations on Graphs. In *Advances in Neural Information Processing Systems*.

Ma, J.; Guo, R.; Wan, M.; Yang, L.; Zhang, A.; and Li, J. 2022b. Learning Fair Node Representations with Graph Counterfactual Fairness. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, 695–703. New York, NY, USA: Association for Computing Machinery.

Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; and Galstyan, A. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6): 1–35.

Pradhan, R.; Zhu, J.; Glavic, B.; and Salimi, B. 2022. Interpretable data-based explanations for fairness debugging. *Proceedings of the 2022 ACM SIGMOD International Conference on Management of Data*.

Rouzrokh, P.; Khosravi, B.; Faghani, S.; Moassefi, M.; Vera Garcia, D. V.; Singh, Y.; Zhang, K.; Conte, G. M.; and Erickson, B. J. 2022. Mitigating bias in radiology machine

learning: 1. Data handling. *Radiology: Artificial Intelligence*, 4(5): e210290.

Takac, L.; and Zabovsky, M. 2012. Data analysis in public social networks. In *International scientific conference and international workshop present day trends of innovations*, volume 1. Present Day Trends of Innovations Lamza Poland.

Tan, K. C.; Khor, E. F.; and Lee, T. H. 2005. *Multiobjective evolutionary algorithms and applications*. Springer Science & Business Media.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *International Conference on Learning Representations*.

Wan, M.; Zha, D.; Liu, N.; and Zou, N. 2023. In-processing modeling techniques for machine learning fairness: A survey. *ACM Transactions on Knowledge Discovery from Data*, 17(3): 1–27.

Wang, N.; Lin, L.; Li, J.; and Wang, H. 2022a. Unbiased Graph Embedding with Biased Graph Observations. In *Proceedings of the ACM Web Conference 2022*, WWW '22, 1423–1433. New York, NY, USA: Association for Computing Machinery.

Wang, Z.; Cao, L.; Feng, L.; Jiang, M.; and Tan, K. C. 2024. Evolutionary Multitask Optimization with Lower Confidence Bound-based Solution Selection Strategy. *IEEE Transactions on Evolutionary Computation*.

Wang, Z.; Cao, L.; Lin, W.; Jiang, M.; and Tan, K. C. 2023a. Robust Graph Meta-Learning via Manifold Calibration with Proxy Subgraphs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(12): 15224–15232.

Wang, Z.; Hong, H.; Ye, K.; Zhang, G.-E.; Jiang, M.; and Tan, K. C. 2023b. Manifold Interpolation for Large-Scale Multiobjective Optimization via Generative Adversarial Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8): 4631–4645.

Wang, Z.; Ye, K.; Jiang, M.; Yao, J.; Xiong, N. N.; and Yen, G. G. 2022b. Solving hybrid charging strategy electric vehicle based dynamic routing problem via evolutionary multi-objective optimization. *Swarm and Evolutionary Computation*, 68: 100975.

Wu, T.; Ren, H.; Li, P.; and Leskovec, J. 2020. Graph information bottleneck. *Advances in Neural Information Processing Systems*, 33: 20437–20448.

Zhang, B. H.; Lemoine, B.; and Mitchell, M. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 335–340.

Zhao, Z.; Jiang, M.; Guo, S.; Wang, Z.; Chao, F.; and Tan, K. C. 2020. Improving Deep Learning based Optical Character Recognition via Neural Architecture Search. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, 1–7.

Zhou, X.; Qin, A. K.; Gong, M.; and Tan, K. C. 2021. A Survey on Evolutionary Construction of Deep Neural Networks. *IEEE Transactions on Evolutionary Computation*, 25(5): 894–912.