Proceedings of the Seventh AAAI/ACM Conference on AI, Ethics, and Society (AIES 2024)

Individual Fairness in Graphs Using Local and Global Structural Information

Yonas Sium[1], **Qi Li**[1], **Kush R. Varshney**[2]
[1]Iowa State University
[2]IBM Research, T. J. Watson Research Center
{yas, qli}@iastate.edu, krvarshn@us.ibm

Abstract

Graph neural networks are powerful graph representation learners in which node representations are highly influenced by features of neighboring nodes. Prior work on individual fairness in graphs has focused only on node features rather than structural issues. However, from the perspective of fairness in high-stakes applications, structural fairness is also important, and the learned representations may be systematically and undesirably biased against unprivileged individuals due to a lack of structural awareness in the learning process. In this work, we propose a pre-processing bias mitigation approach for individual fairness that gives importance to local and global structural features. We mitigate the local structure discrepancy of the graph embedding via a locally fair PageRank method. We address the global structure disproportion between pairs of nodes by introducing truncated singular value decomposition-based pairwise node similarities. Empirically, the proposed pre-processed fair structural features have superior performance in individual fairness metrics compared to the state-of-the-art methods while maintaining prediction performance.

Introduction

Graph neural networks (GNNs) are powerful for graph representation learning and find usage in data mining, machine learning, and optimization. Existing GNN methods comprehensively integrate the information of the neighboring nodes to get the final representation of a given node. For example, as shown in Figure 1, the final representation of each node is the aggregation of the neighboring information and its own features. However, the representation learned by GNNs may be systematically and undesirably biased toward certain individuals due to their privileged structural positioning in the network. Imagine a person who is otherwise equally qualified getting a job over another person either because they are from the same ethnic group as the hiring manager (attribute-related bias) *or* because they have friends and acquaintances in common (structure-related bias) as indicated in Figure **??**. Such biases may increase societal stereotypes in predictions (Du et al. 2020).

Recently, there has been much research interest in algorithmic fairness in graph mining. Most existing methods tend to obtain fair graph representations across *groups* in the population (Lambrecht and Tucker 2019; Kamishima and Akaho 2017; Saxena, Fletcher, and Pechenizkiy 2022; Liu, Nguyen, and Fang 2023; Dong et al. 2022; Dai and Wang 2021; Agarwal, Lakkaraju, and Zitnik 2021; Loveland et al. 2022). Group fairness requires that the algorithm not yield discriminatory predictions or decisions on average within a specific subgroup defined by protected attributes (Dwork et al. 2012). Another notion of fairness is *individual fairness* (Wasilewski and Hurley 2016), which aims for similar outcomes for similar individuals. This paper focuses on **graph individual fairness** rather than graph group fairness.

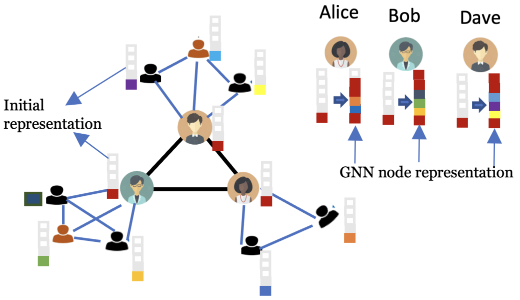


Figure 1: Differences in structural properties (such as no. of links) can lead to varying output node embeddings for similar individuals.

Some existing bias mitigation algorithms for graph individual fairness are *in-processing* methods (Dong et al. 2021; Kang et al. 2020) that enforce fairness during training through constraints or regularization terms in the training algorithm. In-processing methods usually need additional regularization computation to enforce fairness. Alternatively, *post-processing* individual fairness algorithms aim to mitigate biases on the embedding of the (biased) trained model (Petersen et al. 2021). However, post-processing may require additional training of another model on the generated embedding with some fairness constraints. Additionally, enforcing fairness through post-processing may not preserve the structural property of the network embedding learned by the GNN model. Importantly, considering node *structural* information, which may yield inconsistent fairness, is essential in preserving the structural property of the graph during graph learning.

Preserving the structural property of the graph has been used in many GNN tasks (Li et al. 2021a, 2023; Huang et al. 2022). Explicit use of local and global structural information of the graph can mitigate bias issues. For instance, students may connect with classmates and family members on social networks. The student's classmates and family members might have different initial features. As illustrated in Figure **??**, the information flow from those two different communities to the student may lead the student's final embedding to be different from his/her classmates' final embedding during GNN learning. But learning node structural feature similarity can reduce the problem of inequality between a pair of student nodes (Liu, Nguyen, and Fang 2023). For example, *Alice*, *Bob*, and *Dave* are classmates; if there is an edge between *Alice* and *Bob*, and between *Bob* and *Dave*, there is a high probability of an edge between *Alice* and *Dave*. In a citation network, a paper with more initial citations could potentially get more future citations compared to another paper of similar quality on the same subject. In general, nodes with higher degrees within the network tend to possess significant advantages, leading to more favorable outcomes than expected. Therefore, providing the structural property of the network can help similar individuals obtain similar output representations.

In this work, we address the challenges of GNN individual fairness by introducing node structural features as a *pre-processing* step. Specifically, we extract *local* structure features through locally fair PageRank (Tsioutsiouliklis et al. 2021) and *global* structure features through truncated singular value decomposition (SVD). Furthermore, to reduce the cost of computing locally fair PageRank, we extract the subgraphs of the top t important nodes as a proxy and compute a fair distribution of PageRank for the nodes of each subgraph. We concatenate the computed fair PageRank and Truncated SVD of the graph and then compute the pairwise similarities for all nodes, which are used as node structural features. Two GNN models are trained on the node structural features and the original features of the nodes, respectively; the final representations are concatenated with a linear layer for prediction.

The major contributions of our work are summarized as follows:

- We propose a graph-based individually fair method called **SFIF**, that considers the potential bias that arises due to the *structural* disproportion of the individual nodes of the graph.
- We propose a novel scalable structure-aware preprocessing procedure to learn node representation for individually fair graph mining. Specifically, we minimize the disproportional distribution of local and global structural information among the nodes in the denser part of the graph, where influential nodes exist.
- We conduct extensive node classification and link prediction experiments on different real-world datasets. The results show that our proposed method outperforms existing baselines on prediction and individual fairness performance.

Preliminaries

Graph Representation Learning Problem

Consider a graph $G = (V, E)$, where $V = \{v_i\}_{i=1}^n$ denotes the set of nodes, E denotes the set of edges, and $X = \{x_i \in \mathbb{R}^d\}_{i=1}^n$ denotes the node features. Let $A \in \mathbb{R}^{n \times n}$ denote the adjacency matrix of the graph G. Graph representation learning aims to generate graph or node embeddings by providing graph-structured data to the graph mining algorithm, such that the generated embedding is used for downstream tasks such as node classification and link prediction. Suppose $X \to Y$ denotes the ground truth association between the features and targets; the learned representation is denoted as Z. The downstream task takes the representation Z and produces prediction labels $\hat{Y}$.

Individual Fairness

The goal of individual fairness (Dong et al. 2023) is that similar individuals should be treated similarly, which may be expressed mathematically as a Lipschitz condition (Dwork et al. 2012) that the similarity between a pair of nodes in the output space is less than or equal to the corresponding scaled similarity in the input space. The most commonly used individual fairness metrics are *consistency* (Zemel et al. 2013) and *ranking-based fairness* (Dong et al. 2021). In our empirical section, we use ranking-based fairness because it is more common in the GNN individual fairness literature. Ranking-based fairness begins with a pairwise oracle similarity matrix among the nodes S_G and a pairwise similarity matrix based on the predicted labels of the downstream task $\hat{Y}$, denoted $S_{\hat{Y}}$. The predictions are individually fair if, for all nodes v_i, ranked lists of similarity of node i to all other nodes in S_G and $S_{\hat{Y}}$ are consistent with each other. More precisely, for each node u_i, if its similarity value to u_j compared to u_m is greater in the $S_{\hat{Y}}$ matrix (where $\hat{s}_{i,j}$ represents the similarity between nodes u_i and u_j), then this higher similarity relationship between u_i and u_j should also hold in the oracle similarity matrix S_G (where $s_{i,j}$ represents the similarity between nodes u_i and u_j). This condition is met when $\hat{s}_{i,j} > \hat{s}_{i,m}$ with $i \neq j \neq m$.

PageRank

PageRank is a classic graph mining algorithm (Brin and Page 1998) for weighting and ranking the nodes of a graph. It takes graph G as input and provides a weighted rank $p \in \mathbb{R}^n$ to each node based on the random walk distribution on the input graph. The random walk is governed by the transition probability matrix P, which is a normalized version of A. In addition, with probability γ, the walk restarts at a node selected according to restart probability vector r. The weighted rank vector is:

$$p^T = (1 - \gamma)p^T P + \gamma r^T. \tag{1}$$

It finds practical use in scenarios like document ranking, individual prioritization, gene, protein, and molecule assessment in real-world contexts (Li, Xing, and Du 2016; Lee et al. 2011; Avrachenkov et al. 2008).

Graph Neural Networks

Graph neural networks learn node representation by iterative aggregation of neighboring node information (Ma et al. 2021). Some of the prominent architectures Graph neural networks (GNN) are GCN (Kipf and Welling 2016), GraphSage (Hamilton, Ying, and Leskovec 2017), graph isomorphism network (GIN) (Xu et al. 2018), and graph attention(GAT) (Veličković et al. 2017). Most of the existing frameworks encode each node $v_i \in V$ with initial vector representation h_v, and update its representation by iterative aggregation of the neighboring vector representation with its own representation.

The GCN model, widely adopted in GNN, performs the operation $H' = D^{-1}AHW$, where H and H' are input and output features, W is a transformation matrix, and D is a diagonal matrix with degrees of nodes where $D[i,i]^{-1} = \frac{1}{deg(i)}$. Locally, for node i, it computes $h'_i = \frac{1}{deg(i)} \sum_{j \in N(i)} Wx_j$, with $N(i)$ denoting neighbors of node i. The larger neighbor information averaged by the node degree provides a more robust and expressive representation for nodes with higher degrees than those with lower degrees during the GNN learning. This can potentially lead the GNN to bias towards locally higher-degree nodes. From this assumption, providing fair structural features helps to boost the representation of the low-degree nodes and reduce unfairness.

Structural Information as Additional Node Features

It has been proven that despite GNNs being powerful in graph mining, they have limitations in recognizing simple graph structures, such as links and cycles (He et al. 2022). To address this problem, providing additional node features as positional encoding through pre-processing increases the representation power of GNNs (Srinivasan and Ribeiro 2019). Motivated by such expressive power of node structural information, in the next section we detail how we promote individual fairness of GNNs as a pre-processing step by introducing fair node structural features in addition to the original node features. The fair node structural features are a similar to a pair of nodes computed from concatenating a locally fair PageRank and truncated SVD.

Proposed Framework

In this section, we introduce a novel pre-processing-based graph individual fairness framework. As shown in Figure 2, our proposed framework mainly includes two components: (1) computing local and global fair structural features and (2) fair graph representation learning.

Locally Fair Structural Features

Capturing the local structure by sampling neighboring relationships of nodes has been used widely (Gu et al. 2016). Random walk is a commonly used method to learn local reachability between nodes. We use a random walk-based locally fair PageRank algorithm to minimize the bias due to the irregular graph structure (Tsioutsiouliklis et al. 2021).

Subgraph Extraction To reduce the computation cost of the locally fair PageRank algorithm, we consider the densely connected areas of the graph by extracting the h-hop subgraphs of the top t ranked nodes. Subgraphs around important nodes are areas of the graph with more structural connections and are expected to be more informative with respect to the rest of the graph structure (Hamilton, Ying, and Leskovec 2017). Therefore, to compute the locally fair PageRank, we consider the h-hop subgraph around the selected s nodes.

We first designate the importance of every node in the graph using the HITS algorithm (Kleinberg 1999) to obtain an authority value vector $\in \mathbb{R}^n$ and hub value vector $\in \mathbb{R}^n$. Nodes with high authority value and nodes with high hub value are considered to be important. For each important node v_i subgraph extraction, let $\mathcal{N}^+(v_i) = \{j|(i,j) \in E\}$ be the set of nodes that v_i points to, $\mathcal{N}^-(v_i)$ be the set of source nodes targeting v_i and $\mathcal{N}(v_i) = \mathcal{N}^+(v_i) \cup \mathcal{N}^-(v_i)$ be the set of all incident nodes. We recursively define $G^i = (V^i, E^i)$, by setting $G^i_0 = (V^i_0, E^i_0) = (\{i\}, \emptyset)$ and $G^i_l = (V^i_l, E^i_l)$ where $V^i_l = \bigcup_{j \in V^i_{l-1}} \mathcal{N}(j)$ and $E^i_l = \{(i,j) \in E^i | i,j \in V^i_l)\}$, for $l = 1, 2, \ldots, t$. We include node v_i in the extracted subgraph.

We extract a h-hop subgraph G^i around a node v_i, where v_i is in the list of the top t important nodes. The h-hop subgraph around a node contains nodes within h-hop and all the edges between these nodes. Then the extracted subgraphs are provided to the locally fair PageRank algorithm to generate a fairly distributed PageRank score for all nodes in the subgraphs.

Locally Fair PageRank Inspired by the broad utilization of PageRank in assessing the specific significance of a node in relation to a local subset of nodes, we compute locally fair PageRank from the extracted subgraph to make each individual node act fairly during the distribution of its PageRank to the neighboring nodes. From the commonly used assumption (Hamilton, Ying, and Leskovec 2017), each node in the subgraph has a low dependency on nodes outside its h-hop subgraph. Therefore, each node in the subgraph is expected to distribute its PageRank fairly to its neighbors. The basic idea is to change the transition probability matrix P to obtain a fair random walk during the PageRank computation. Each node in the subgraph jumps with the probability of ϕ to the low-ranked nodes of the subgraph.

Initially, we group the nodes in the subgraph into high and low ranked groups of nodes based on the computed authority and hub values using the HITS algorithm. The lowest-ranked S nodes of the given subgraph are considered structurally protected nodes. The PageRank is asked for a fair distribution of weights to the protected nodes, which are considered structurally unprivileged. For fairness, each node distributes a fraction ϕ of its PageRank to the protected nodes and distributes $1 - \phi$ to the rest of the unprotected nodes. ϕ is computed as the ratio of the number of nodes in the structurally underrepresented $|S|$ and the total number of nodes in the subgraph $|V_t|$, $\phi = \frac{|S|}{|V_t|}$. We provide $\frac{1}{|V|}$ PageRank value for nodes outside the extracted subgraphs. We formulate our

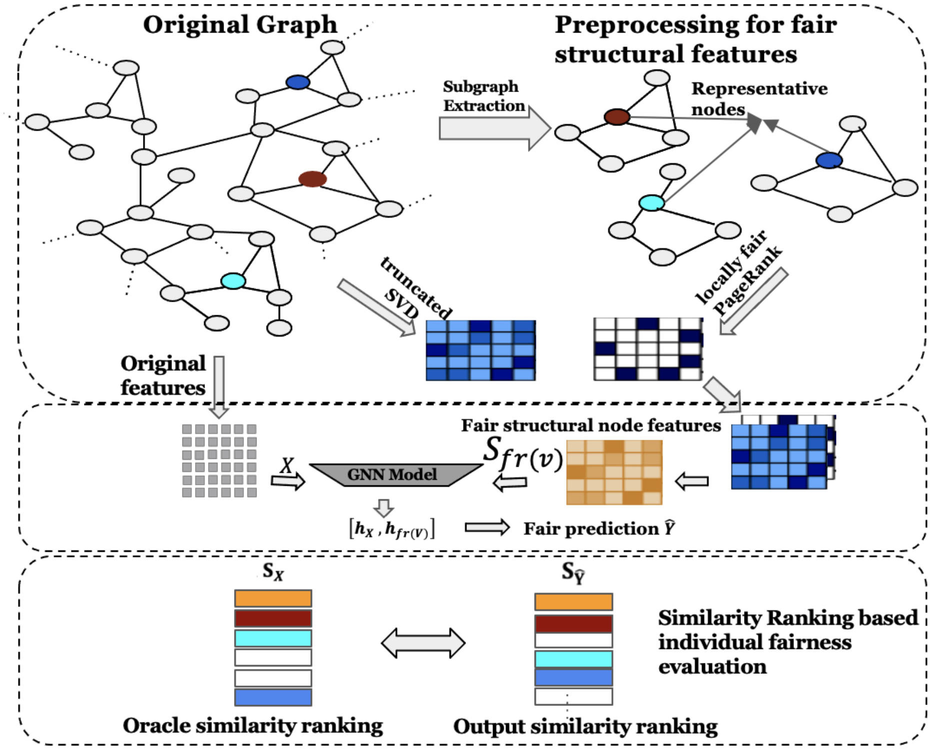

Original Graph
Preprocessing for fair structural features
Subgraph Extraction
Representative nodes
truncated SVD
locally fair PageRank
Original features
Fair structural node features
$S_{fr(v)}$
X
GNN Model
$[h_X, h_{fr(V)}]$
Fair prediction $\hat{Y}$
S_X
$S_{\hat{Y}}$
Similarity Ranking based individual fairness evaluation
Oracle similarity ranking
Output similarity ranking

Figure 2: An Overview of the Proposed Framework

locally fair PageRank distribution as follows.

The jump vector for managing the transition matrix for the S nodes is defined as:

$$P_S[i,j] = \begin{cases} \frac{1}{|S|}, & \text{if } (i,j) \text{ belong to } E \text{ and } j \in S \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Similarly, we define the jump vector probability to the remaining $V_t - S$ nodes. Then, the transition matrix is defined as:

$$\text{Transition matrix} = \phi P_S + (1-\phi) P_{V_t - S} \tag{3}$$

Finally, we computed the locally fair features of each node by using the optimized PageRank as follows:

$$PR[v] = \sum_{i=1}^{K} \phi \cdot \frac{PR[u_i]}{\text{degree}[u_i]}, \quad \text{where } u_i \text{ is a neighbor of } v. \tag{4}$$

Definition 1 (Locally Fair Structural Features of PageRank)**.** *For a given subgraph $G_t = (V_t, E_t)$, a subset of nodes $S \subset V_t$ are structurally underrepresented. A PageRank algorithm 1, $PR \in PR(G_t)$ is locally ϕ-fair on S nodes, if for all rows of the transition matrix P, every node $v_i \in V_t$ in the subgraph $\sum_{j \in S} P_{i,j} = \phi$ and the restart jump vector r is ϕ-fair if $\sum_{j \in S} v_j = \phi,\ \phi \in (0,1)$.*

The locally fair PageRank outlined in Algorithm 1 is ϕ-fair by granting fairness at every step of the random walk. Such behavior helps to achieve global PageRank fairness. Let's denote the computed locally fair structural feature

Algorithm 1: Locally Fair Structural Feature from Optimization of PageRank

```
Input: $A$: adjacency matrix of the input graph $G$;
$\alpha$: damping factor;
$\epsilon$: convergence threshold;
$n$: number of nodes in $G$;
$index$: a list of nodes with the least importance score;
Output: Locally fair structural feature
while convergence = false do
    for each node $v$ in $G$ do
        $newPR(v) = (1-\phi)/n$;
        for each node $u$ linked to $v$ do
            update $\phi$ using $u's$ uniform distribution
             $PR$ to its neighbors following sec. ;
            $newPR[v]\ += $
             $\phi * (oldPR[u]/degree[u])$;
    $convergence = true$;
    for each node $v$ in $G$ do
        if $|newPR[v] - PR[v]| > \epsilon$ then
            $convergence = false$;
            break;
    $PR = newPR$;
return $PR$
```

from the optimization of PageRank for a subgraph G_t as $LFPR[V_t]$.

Global Structural Features

Graph-level structural features help GNNs learn clean graph structures. Singular value decomposition (SVD) based embedding preserves the higher order proximity between the nodes (Paige and Saunders 1981), allowing us to capture long-range dependence between nodes. SVD is a matrix factorization denoted by $A = U\Sigma V^T$. To reduce the dimensionality space for large-scale graphs, we use truncated SVD, an approximate variant of SVD. The truncated SVD of the adjacency matrix $A(u,v) \in \mathbb{R}^{n\times n}$ is $U_d \in \mathbb{R}^{n\times d}$ of its left singular subspace corresponding to its top d singular values, Σ is an $n \times n$ diagonal matrix containing singular values and V is an $n \times n$ orthogonal matrix containing right singular vectors.

Theorem 1. *Let G be a graph with V vertices and E edges. If L is the Laplacian of G, the singular value decomposition captures the global structural information of the graph.*

Proof. Let $L(G) = D - A$ where D is the diagonal matrix and A is the adjacency matrix. Let SVD of $L(G) = U\Sigma V^T$. Let $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$ be the singular values ordered in non-increasing order.
Claim 1: The largest singular value σ_1 corresponds to the most significant structural component of the graph, and the d largest singular values, $\sigma_1, \sigma_2, \ldots, \sigma_d$, collectively correspond to the most prominent global structural features of the graph, which are the d left singular vectors $u_1, u_2, \ldots, u_d$.
Proof of Claim 1: Consider the matrix $L(G) = U\Sigma V^T$. Let x be the first right singular vector corresponding to σ_1, i.e., $V^T_{\cdot,1} = x$. Therefore, $L(G) \times x = U\Sigma V^T \times x = U\Sigma e_1 = \sigma_1 u_1$, where e_1 is the first canonical basis vector.

This shows that σ_1 captures the importance of the dominant structural feature of the graph u_1. By considering the d largest singular values, $\sigma_1, \sigma_2, \ldots, \sigma_d$, and their corresponding singular vectors, $u_1, u_2, \ldots, u_d$, we obtain the d most significant global structural features in the graph.
Robustness to Structural Changes: The dominant singular values and vectors are likely to change for any structural change in the graph(e.g., node or edge addition/removal).

Therefore, from the above claim and proof presented, we conclude that SVD captures global structural information of the graph.

□

Combined Fair Structural Features

After computing the locally fair PageRank and the truncated SVD as a fairness pre-processing step, we concatenate them as:

$$fr(V) = \big[LFPR[V], \quad \text{truncated-}SVD[V]\big]. \tag{5}$$

The pair-wise cosine similarities of nodes on the computed fair structural features $fr(V)$ are used as additional node features to promote fairness during graph mining. The main idea of learning fair structural similarity between a pair of nodes is to maintain the structural-semantic similarity by minimizing the potential discrepancy of the embedding between two similar nodes due to the additional information aggregation from the nonessential neighboring nodes during the GNN learning.

Datasets	Nodes	Edges	Features	Classes
ACM	16,484	71,980	8,337	9
CS	16,484	71,980	8,337	9
Phy	34,493	247,962	8,415	5
BlogCatalog	5,196	171,743	8,189	N/A
Flickr	7,575	239,738	12,047	N/A
Facebook	4,039	88,234	1,406	N/A

Table 1: Dataset statistics

Fair Node Representation Learning

As described in Section and Figure 2, the extracted fair structural features ($fr(V)$) are used as an additional input to the original node feature (X) to promote fairness in downstream tasks of the GNN. To improve the expressive power of the GNN, we train the fair structural features and original features in parallel such that one can not mask the other during training. Finally, we concatenate the representation from the fair structural features ($h_{fr(V)}$) and the original node features (h_X) for prediction. Cross-entropy loss is used to evaluate the downstream task, such as node classification or link prediction.

Experiments

Datasets

For node classification tasks, we use ACM citation network (Tang et al. 2008) and co-authorship networks of Co-author-CS and Co-author-Phy (Shchur et al. 2018) . For the ACM citation networks, each node represents a paper, the edges indicate the citation relationship between two papers, and the bag of words of the abstract of each paper are the features for each node. For Co-author-CS and Co-author-Phy networks, the nodes indicate the authors and the edges indicate the co-author relationship for a paper. The bag of words of the paper's abstract is the node features.

For link prediction, we use BlogCatalog (Tang and Liu 2009), Facebook (Leskovec and Mcauley 2012), and Flickr (Huang, Li, and Hu 2017) datasets. The nodes of these networks are users. The edges are interactions between users. The node features are the bag of words of the user profile (Dong et al. 2021).

The detailed statistics of these datasets are provided in Table 1, including the number of nodes, the number of edges, the dimension of their features, and the number of classes.

Experiment Settings

Baselines. We compare our proposed method with two categories of methods: 1) GNN models using the initial node features, and 2) GNN models with state-of-the-art individual fairness methods. For the first categories, we compare with graph convolutional network (GCN) (Kipf and Welling 2016), graph isomorphism network (GIN) (Xu et al. 2018),

Dataset	Model	ACC	NDCG@10(structural)	NDCG@10(feature)
ACM	GCN	71.27 ± 0.64	24.52 ± 0.92	45.20 ± 1.283
	GAT	77.25 ± 0.09	25.98 ± 0.18	45.20 ± 1.28
	GIN	79.52 ± 0.98	28.59 ± 0.27	44.41 ± 1.57
	SAGE	77.32 ± 0.75	26.51 ± 1.28	42.25 ± 0.71
	REDRESS(GCN)	72.01 ± 0.75	32.04 ± 1.08	**48.89 ± 0.12**
	INFORM(GCN)	68.37 ± 0.52	15.78 ± 0.76	40.01 ± 0.82
	SFIF(GCN)(ours)	**77.46 ± 0.23**	**69.21 ± 0.74**	45.43 ± 0.27
	SFIF(GAT)(ours)	76.89 ± 0.98	62.58 ± 0.43	41.92 ± 0.47
	SFIF(GIN)(ours)	78.79 ± 0.62	66.65 ± 0.21	42.25 ± 0.52
	SFIF(SAGE)(ours)	75.61 ± 1.02	68.84 ± 0.32	44.43 ± 0.22
Co-author-CS	GCN	87.93 ± 1.02	21.59 ± 1.37	52.72 ± 0.53
	GAT	89.69 ± 0.67	20.76 ± 0.65	50.31 ± 1.09
	GIN	90.43 ± 2.01	24.22 ± 1.34	53.13 ± 0.98
	SAGE	88.17 ± 1.09	22.71 ± 0.82	50.49 ± 1.23
	REDRESS(GCN)	88.72 ± 2.92	28.42 ± 0.96	58.42 ± 0.13
	INFORM(GCN)	88.91 ± 1.06	15.78 ± 0.17	52.73 ± 0.27
	SFIF(GCN)(ours)	**92.47 ± 1.52**	**73.29 ± 0.57**	**59.66 ± 0.34**
	SFIF(GAT)(ours)	91.43 ± 1.02	67.93 ± 0.29	58.18 ± 0.47
	SFIF(GIN)(ours)	93.83 ± 0.52	73.42 ± 0.15	61.67 ± 0.17
	SFIF(SAGE)(ours)	89.72 ± 0.14	74.12 ± 0.55	61.02 ± 0.22
Co-author-Phy	GCN	93.73 ± 1.42	3.31 ± 0.75	37.22 ± 0.24
	GAT	90.19 ± 1.07	2.54 ± 1.34	38.71 ± 0.76
	GIN	92.13 ± 2.01	4.98 ± 0.23	40.35 ± 1.21
	SAGE	91.18 ± 1.91	4.37 ± 0.91	38.59 ± 0.92
	REDRESS(GCN)	92.51 ± 0.30	5.92 ± 0.29	**44.92 ± 0.91**
	INFORM(GCN)	93.01 ± 0.75	2.31 ± 0.71	34.99 ± 0.28
	SFIF(GCN)(ours)	**95.17 ± 0.95**	**49.24 ± 0.96**	40.79 ± 0.55
	SFIF(GAT)(ours)	92.18 ± 1.02	54.08 ± 0.69	43.09 ± 0.82
	SFIF(GIN)(ours)	93.10 ± 1.72	56.08 ± 0.23	46.09 ± 0.82
	SFIF(SAGE)(ours)	90.11 ± 1.97	54.85± 0.35	44.96 ± 0.74

Table 2: Comparison of the performance of node representation learning methods with respect to node classification prediction and individual fairness on ACM, Co-author-CS, and Co-author-Phy(Phy) datasets. ACC indicates node classification accuracy performance. NDCG@k(structural) and NDCG@k(feature) refer to average ranking individual fairness based on structural and initial node features, respectively. Our proposed method is denoted as structural features for individual fairness (SFIF). The best values of prediction and individual fairness with GCN as the backbone model are bolded. The interval after the metric value reports the standard deviation based on 4 runs.

GraphSage (Hamilton, Ying, and Leskovec 2017) and graph attention (GAT) (Veličković et al. 2017). For GNNs with individual fairness constraints, we compare with REDRESS (Dong et al. 2021) and InFoRM (Kang et al. 2020) with GCN as the backbone GNN models. REDRESS is a rank-based optimization method that tries to maintain the relative similarity ranking between a given instance node and other sample nodes. The ranking is computed based on the similarity matrix for a pair of nodes relative to the instance node before and after the GNN representation learning. InFoRM is an optimization method based on the Lipschitz condition on traditional graph mining tasks such as spectral clustering and PageRank, but it is not designed for GNN. We adopt the REDRESS approach to modify InFoRM method by combining the fairness loss and the utility loss together to improve the individual fairness of the GNN model.

Loss for Classification: In the context of node classification, the final layer of the Graph Neural Network (GNN) usually adjusts its dimensionality to match the number of classes. It employs a softmax activation function, where the i^{th} output dimension corresponds to the probability of class i. We optimize the model's parameters by minimizing the cross-entropy loss, which is defined as follows:

$$L = -\frac{1}{|Y|}\sum_{i}\sum_{j} Y_{i,j}\log(\hat{Y}_{i,j}) \tag{6}$$

Where $|Y_{i,j}|$ represents the number of rows in $Y_{i,j}$, which corresponds to the number of labeled examples, and $\hat{Y}_{i,j}$ represents the probability of the i^{th} example belonging to the j^{th} class.

For link prediction, the objective of the training is to minimize the cross entropy between the reconstructed adjacency matrix and the true adjacency matrix.

$$L = \sum_{i\in V}\sum_{j\in V}(-A_{i,j}\log \hat{A}_{i,j} - (1-A_{i,j})\log(1-\hat{A}_{i,j})) \tag{7}$$

Dataset	Model	AUC	NDCG@10(structural)	NDCG@10(feature)
BlogCatalog	GCNLink	85.81 ± 0.75	31.89 ± 0.21	**17.73 ± 0.37**
	GATLink	86.78 ± 0.67	30.87 ± 0.39	16.75 ± 0.82
	GINLink	86.78± 0.29	31.32 ± 0.28	18.01 ± 0.56
	SAGELink	86.42 ± 0.39	32.21 ± 0.62	18.12 ± 0.88
	REDRESS(GCNLink)	**86.59 ± 0.51**	34.12 ± 0.75	17.66 ± 0.92
	INFORM(GCNLink)	27.17 ± 0.35	27.01 ± 0.23	16.51 ± 0.71
	SFIF(GCNLink)(ours)	86.49 ± 0.27	**38.89 ± 0.58**	16.47 ± 0.34
	SFIF(GATLink)(ours)	85.34 ± 0.25	37.21 ± 0.36	16.72 ± 0.72
	SFIF(GINLink)(ours)	87.27 ± 0.37	38.51 ± 0.27	17.24 ± 0.78
	SFIF(SAGELink)(ours)	87.15 ± 0.77	40.12 ± 1.05	18.25 ± 0.21
Facebook	GCNLink	96.69 ± 0.73	17.58 ± 0.28	23.27 ± 1.02
	GATLink	95.55 ± 0.74	17.29 ± 0.82	24.57 ± 0.79
	GINLink	97.12 ± 0.39	17.52 ± 0.32	25.13 ± 0.55
	SAGELink	95.88 ± 0.38	18.22 ± 1.23	26.71 ± 0.85
	REDRESS(GCNLink)	96.49 ± 0.37	29.67 ± 0.59	27.92 ± 0.47
	INFORM(GCNLink)	92.12 ± 1.35	17.19 ± 0.25	**29.65 ± 0.28**
	SFIF(GCNLink)(ours)	**98.89 ± 0.35**	**60.01 ± 0.65**	28.79 ± 0.71
	SFIF(GATLink)(ours)	97.34 ± 0.38	58.01 ± 0.88	27.88 ± 0.51
	SFIF(GINLink)(ours)	98.91 ± 0.81	59.22 ± 0.39	28.25 ± 0.36
	SFIF(SAGELink)(ours)	97.28 ± 0.75	60.59 ± 0.71	29.55 ± 0.77
Flickr	GCNLink	91.22 ± 0.39	22.31 ± 0.79	14.24 ± 0.84
	GATLink	92.33 ± 0.77	21.81 ± 0.71	13.12 ± 0.77
	GINLink	92.87 ± 0.49	22.15 ± 0.51	14.29 ± 0.81
	SAGELink	92.17 ± 0.28	23.12 ± 0.33	13.52 ± 0.69
	REDRESS(GCNLink)	**93.67 ± 0.35**	28.76 ± 0.38	14.13 ± 0.47
	INFORM(GCNLink)	90.17 ± 0.76	22.89 ± 0.39	14.58 ± 0.71
	SFIF(GCNLink)(ours)	91.89 ± 0.28	**43.25 ± 0.61**	**16.61 ± 0.51**
	SFIF(GATLink)(ours)	92.01 ± 0.81	41.71 ± 0.66	15.12 ± 0.97
	SFIF(GINLink)(ours)	93.25 ± 0.39	40.82 ± 0.73	15.55 ± 0.49
	SFIF(SAGELink)(ours)	91.94 ± 0.58	42.28 ± 0.79	16.13 ± 0.48

Table 3: Comparison of the performance of node representation learning methods with respect to link prediction and individual fairness on Blogcatalog, Facebook and Flicker datasets. AUC, indicates to the area under the curve link prediction performance. NDCG@k(structural) and NDCG@k(feature) refer to average ranking individual fairness based on structural and initial node features, respectively. Our proposed method is denoted as structural features for individual fairness (SFIF). The best values of prediction and individual fairness with GCN as the backbone model are bolded. The interval after the metric value reports the standard deviation based on 4 runs.

where $A_{i,j} = 1$ represents positive edges and $A_{i,j} = 0$ represents negative edges.

Evaluation metric. We evaluate the proposed method in terms of both model prediction performance and fairness towards individual nodes. To measure the prediction performance, we use accuracy (ACC) for node classification and area under the curve (AUC) for link prediction. To measure the individual fairness of the node embedding generated by the models, we adopt the widely used ranking metric NDCG@k (Järvelin and Kekäläinen 2002).

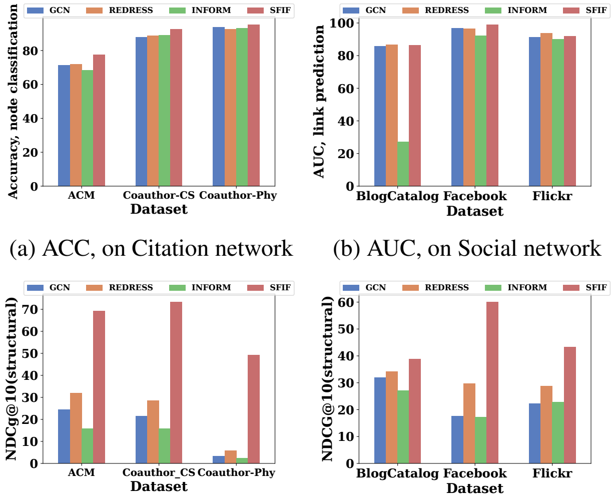
NDCG@k for individual fairness operates on the similarity between the ranked list of the outcome similarity matrix $S_{\hat{Y}}$ and the oracle (input) similarity matrix S_G for each node, with the underlying principle that higher ranked nodes in the list should carry more weight than lower ranked ones. We compute the oracle similarity S_G of structure from the adjacency matrix using Jaccard similarity and the oracle similarity S_G of attributes from the original node features using cosine similarity. For the output embedding similarity $S_{\hat{Y}}$, we use cosine similarities for both structural and node features. To compute NDCG@k, we divide the discounted cumulative gain (DCG) of the ranked list by the ideal DCG (IDCG), where nodes in the list are ordered in the most optimal manner based on their initial input similarity in S_G. Mathematically, it can be formulated as:

$$NDCG@k = \frac{DCG@k}{IDCG@k}, \text{where } DCG@k = \sum_{i=1}^{k} DG(p). \quad (8)$$

Here DG(p) is the discounted relative similarity score gain at position p in the list with k nodes, and IDCG@k is the ideal DCG@k. To get the final NDCG@k individual fairness performance, we take the average NDCG@k of all nodes.

$$Average\ NDCG@k = \sum_{i=1}^{n} NDCG@k_i/|V| \quad (9)$$

For the average ranking value of NDCG@k, we studied different values of k, and the average ranking value improves for both structural and attribute features with the increase of k, and the relative ordering of methods remains constant. For simplicity of exposition, we report results for k = 10. For higher k = 25 is reported in the appendix.



(a) ACC, on Citation network (b) AUC, on Social network

(c) NDCG@10, on Citation net (d) NDCG@10, on Social net

Figure 3: Performance comparison between SFIF and baselines on utility (node classification; (ACC), link prediction (AUC)), and individual fairness (NDCG@10).

- The individual fairness NDCG@10 score for **SFIF** is much larger than the baselines, as indicated in figure 3c and figure 3d. This is because the provided fair structural feature preserves the structural property of the input graph during the GNN learning.
- The performance of **SFIF** for node classification (ACC) and link prediction (AUC) is much higher than the baselines in most of the datasets, as shown in Figure 3a and Figure 3b. This shows the provided fair structural feature helps to improve the expressive representation of GNN's while preserving individual fairness.

Fairness and Prediction Performance

The model's prediction performance for node classification and ranking-based individual fairness is shown in Table 2. The link prediction model performance and individual fairness results are shown in Table 3. The bold font indicates the best-performing method for GCN backbone GNN model. Our results are summarized as follows:

We compare our proposed method with several GNN models from the structural and feature perspectives. From a structural standpoint, we want nodes with many common neighbors to have similar encodings after GNN to preserve the structural property. From the feature standpoint, we want the nodes with similar features to still have similar node embedding after GNN. For node classification and link prediction, for all datasets, our method consistently achieves the best performance under structural individual fairness in NDCG@10 average ranking score. For example, our method outperforms GCN by $\{44.69, 51.70, 45.93\}$ individual fairness for structural oracle similarity in node classification. Furthermore, our method achieves comparable performance individual fairness for the original node feature oracle similarity.

We also compare with baselines that involve individual fairness methods. Our method still significantly outperforms these fairness methods in terms of structural individual fairness in NDCG@10 average ranking score for all datasets. For Example, our method outperforms REDRESS by $\{37.17, 64.87, 43.32\}$ individual fairness for structural oracle similarity in node classification. This could be mainly because our method involves the local and global structural similarity between node pairs that helps preserve the original network structure patterns. Furthermore, our method outperforms those baseline methods in downstream task performance. Even though the existing baseline methods are mainly designed to optimize the original feature to achieve fairness, our method achieves comparable individual fairness in NDCG@10 for original node feature oracle similarity.

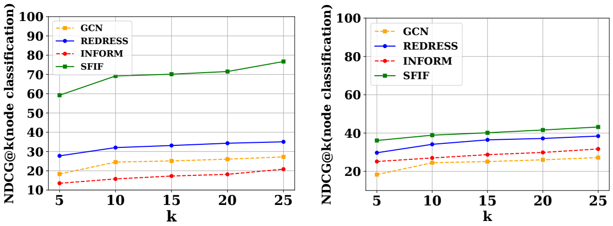
Significant works (Srinivasan and Ribeiro 2019) have shown that structural features are more prevalent than original node features for link prediction tasks. This could be one reason that link prediction tasks benefit more from our fair structural features for structural individual fairness compared to the node feature individual fairness. For example, our method outperforms GCNLink by $\{7.00, 42.43, 20.94\}$ for structural individual fairness and $\{-1.26, 6.38, 2.37\}$ for node feature individual fairness. Therefore, those observations validate the effectiveness of our method.

Datasets	Structural Features	ACC	NDCG@10 (structural)	NDCG@10 (feature)
ACM	LFPR	**77.87**	57.00	40.79
	truncated-SVD	76.67	70.17	43.66
	SFIF	77.46	**71.29**	**45.43**
Co-author-CS	LFPR	**93.78**	67.66	59.36
	truncated-SVD	92.25	61.36	67.05
	SFIF	92.47	**69.17**	**69.66**
Co-author-Phy	LFPR	95.35	46.36	48.20
	truncated-SVD	**96.17**	48.59	**50.46**
	SFIF	95.17	**49.24**	47.79

Table 4: Ablation study on node classification and individual fairness for locally fair Pagerank(LFPR) and truncated SVD

Ablation Study

We conduct an ablation study to verify the effectiveness of the computed structural features using the locally fair PageRank and the truncated SVD. First, we remove the features computed using the locally fair PageRank and only use features computed using truncated SVD for pairs of node similarity as pre-processing fairness-aware structural node features. Next, we remove truncated SVD features and use node pair similarity from the locally fair PageRank features as our fairness-aware pre-processing fair structural node feature. The prediction performance and ranking-based individual fairness are shown in Table 4. We observe that the fairness performance gets reduced in all variants than when used together. We investigate the effect of different k from a selection of $[5, 10, 15, 20, 25]$ on the model's individual fairness performance. Figure 4a and Figure 4b shows, for both node classification and link prediction, the larger the value of k, the better the individual fairness performance for both SFIF and the baselines. For different variants of k value, SFIF consistently performs better than the baselines on individual fairness without affecting the utility performance of the model.



(a) ACM

(b) BlogCatalog

Figure 4: Individual fairness SFIF performance with different choices of k.

Related Work

Techniques used in fairness aware graph mining. Due to the wide usage of graph machine learning systems in many high-stakes applications, such as loan approval systems and recommender systems, researchers have performed significant work to measure the fairness of those systems (Wang et al. 2019; Fu et al. 2020; Leonhardt, Anand, and Khosla 2018; Navarin, Oneto, and Donini 2020; Kleindessner et al. 2019; Horowitz and Kamvar 2010). Fairness can broadly be categorized into group fairness, individual fairness, and counterfactual fairness. Various works try to improve the unfair graph mining algorithm via different techniques such as **optimization with regularization** to encourage the fairness level of the algorithm output (Franco et al. 2022; Fan et al. 2021; Burke et al. 2017), **adversarial learning** where the discriminator tries to predict the sensitive features from the embedding generated by the generator (Bose and Hamilton 2019; Xu et al. 2021; Arduini et al. 2020), and **edge rewiring** to modify the network topology by adding or removing some edges to balance the structural information flow between the nodes (Li et al. 2021b; Laclau et al. 2021; Jalali et al. 2020).

Individual fairness in graph mining. A few studies have been done on individual fairness. However, these studies are mainly focused on optimizing the potential bias that emerged due to the causal influence effect of sensitive features of the neighboring nodes during information aggregation as the in-processing approach. Recent works (Kang et al. 2020; Dong et al. 2021) on individual fairness focus on in-processing to mitigate fairness via regularization in the form of $L = L_{utility} + \lambda L_{fair}$, where $L_{utility}$ and λL_{fair} are the objectives of the downstream task and fairness, respectively, and λ is a hyperparameter that controls the level of fairness. These models require additional model training to promote fairness. For example, REDRESS (Dong et al. 2021) proposes a ranking-based individual fairness optimization method to encourage each individual node relative similarity ranking list to the other nodes during the input and output.

Group fairness in graph mining. Group fairness emphasizes equitable treatment of subgroups with respect to some sensitive attribute defining groups. FairGNN (Dai and Wang 2021) and UGE (Wang et al. 2022) proposed a group fairness methods based on the learned node representations. NIFTY (Agarwal, Lakkaraju, and Zitnik 2021) incorporates a counterfactual group fairness method for nodes and edges to reduce bias. EDITS (Dong et al. 2022) introduced both attribute and structural debiasing to mitigate group fairness. Deg-FairGNN (Liu, Nguyen, and Fang 2023) aims to ensure fair outcomes for nodes with varying degrees within a group. Despite EDITS and Deg-FairGNN introduced structural fairness, they are designed for group fairness and cannot be extended for individual fairness.

Summary and Discussion

In this paper, we propose a novel approach for graph individual fairness that considers the bias due to irregular graph structure and learning heterogeneous features from neighboring nodes by preserving the structural property between the graph nodes. We propose novel fairness-aware structural features through a pre-processing step to learn GNN node representations that can scalably achieve graph individual fairness and improve prediction performance. Specifically, we use a locally fair PageRank to learn the local structural property of the nodes fairly. Furthermore, we learn global structural awareness of the nodes via truncated SVD. Finally, we couple the two structural features and train a GNN with these as additional fairness-preserving node features. Our proposed fairness-aware structural features maximize the similarity between the learned node representation and the original data. Experiment results on real-world graph data validate the effectiveness of our proposed framework for task prediction and individual fairness.

Acknowledgments

The work is supported in part by the US National Science Foundation under grant No.2152117 and IIS-2007941.

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.

Arduini, M.; Noci, L.; Pirovano, F.; Zhang, C.; Shrestha, Y. R.; and Paudel, B. 2020. Adversarial learning for debiasing knowledge graph embeddings. *arXiv preprint arXiv:2006.16309*.

Avrachenkov, K.; Dobrynin, V.; Nemirovsky, D.; Pham, S. K.; and Smirnova, E. 2008. Pagerank based clustering of hypertext document collections. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 873–874.

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

Brin, S.; and Page, L. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7): 107–117.

Burke, R.; Sonboli, N.; Mansoury, M.; and Ordoñez-Gauger, A. 2017. Balanced neighborhoods for fairness-aware collaborative recommendation.

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.

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

Dong, Y.; Liu, N.; Jalaian, B.; and Li, J. 2022. Edits: Modeling and mitigating data bias for graph neural networks. In *Proceedings of the ACM web conference 2022*, 1259–1269.

Dong, Y.; Ma, J.; Wang, S.; Chen, C.; and Li, J. 2023. Fairness in graph mining: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Du, M.; Yang, F.; Zou, N.; and Hu, X. 2020. Fairness in deep learning: A computational perspective. *IEEE Intelligent Systems*, 36(4): 25–34.

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.

Fan, W.; Liu, K.; Xie, R.; Liu, H.; Xiong, H.; and Fu, Y. 2021. Fair graph auto-encoder for unbiased graph representations with wasserstein distance. In *2021 IEEE International Conference on Data Mining (ICDM)*, 1054–1059. IEEE.

Franco, D.; Navarin, N.; Donini, M.; Anguita, D.; and Oneto, L. 2022. Deep fair models for complex data: Graphs labeling and explainable face recognition. *Neurocomputing*, 470: 318–334.

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.

Gu, S.; Lillicrap, T.; Ghahramani, Z.; Turner, R. E.; and Levine, S. 2016. Q-prop: Sample-efficient policy gradient with an off-policy critic. *arXiv preprint arXiv:1611.02247*.

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

He, X.; Hooi, B.; Laurent, T.; Perold, A.; LeCun, Y.; and Bresson, X. 2022. A Generalization of ViT/MLP-Mixer to Graphs. *arXiv preprint arXiv:2212.13350*.

Horowitz, D.; and Kamvar, S. D. 2010. The anatomy of a large-scale social search engine. In *Proceedings of the 19th international conference on World wide web*, 431–440.

Huang, N.; Villar, S.; Priebe, C. E.; Zheng, D.; Huang, C.; Yang, L.; and Braverman, V. 2022. From Local to Global: Spectral-Inspired Graph Neural Networks. *arXiv preprint arXiv:2209.12054*.

Huang, X.; Li, J.; and Hu, X. 2017. Label informed attributed network embedding. In *Proceedings of the tenth ACM international conference on web search and data mining*, 731–739.

Jalali, Z. S.; Wang, W.; Kim, M.; Raghavan, H.; and Soundarajan, S. 2020. On the information unfairness of social networks. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, 613–521. SIAM.

Järvelin, K.; and Kekäläinen, J. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4): 422–446.

Kamishima, T.; and Akaho, S. 2017. Considerations on recommendation independence for a find-good-items task.

Kang, J.; He, J.; Maciejewski, R.; and Tong, H. 2020. Inform: Individual fairness on graph mining. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 379–389.

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

Kleinberg, J. M. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5): 604–632.

Kleindessner, M.; Samadi, S.; Awasthi, P.; and Morgenstern, J. 2019. Guarantees for spectral clustering with fairness constraints. In *International Conference on Machine Learning*, 3458–3467. PMLR.

Laclau, C.; Redko, I.; Choudhary, M.; and Largeron, C. 2021. All of the fairness for edge prediction with optimal transport. In *International Conference on Artificial Intelligence and Statistics*, 1774–1782. PMLR.

Lambrecht, A.; and Tucker, C. 2019. Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads. *Management science*, 65(7): 2966–2981.

Lee, I.; Blom, U. M.; Wang, P. I.; Shim, J. E.; and Marcotte, E. M. 2011. Prioritizing candidate disease genes by network-based boosting of genome-wide association data. *Genome research*, 21(7): 1109–1121.

Leonhardt, J.; Anand, A.; and Khosla, M. 2018. User fairness in recommender systems. In *Companion Proceedings of the The Web Conference 2018*, 101–102.

Leskovec, J.; and Mcauley, J. 2012. Learning to discover social circles in ego networks. *Advances in neural information processing systems*, 25.

Li, H.; Wang, B.; Cui, L.; Bai, L.; and Hancock, E. R. 2021a. LGL-GNN: Learning Global and Local Information for Graph Neural Networks. In *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshops, S+ SSPR 2020, Padua, Italy, January 21–22, 2021, Proceedings*, 129–138. Springer.

Li, P.; Wang, Y.; Zhao, H.; Hong, P.; and Liu, H. 2021b. On dyadic fairness: Exploring and mitigating bias in graph connections. In *International Conference on Learning Representations*.

Li, P.; Yu, H.; Luo, X.; and Wu, J. 2023. LGM-GNN: A local and global aware memory-based graph neural network for fraud detection. *IEEE Transactions on Big Data*.

Li, X.-Q.; Xing, T.; and Du, D. 2016. Identification of top-ranked proteins within a directional protein interaction network using the PageRank algorithm: applications in humans and plants. *Current Issues in Molecular Biology*, 20(1): 13–28.

Liu, Z.; Nguyen, T.-K.; and Fang, Y. 2023. On Generalized Degree Fairness in Graph Neural Networks. *arXiv preprint arXiv:2302.03881*.

Loveland, D.; Pan, J.; Bhathena, A. F.; and Lu, Y. 2022. Fairedit: Preserving fairness in graph neural networks through greedy graph editing. *arXiv preprint arXiv:2201.03681*.

Ma, Y.; Liu, X.; Shah, N.; and Tang, J. 2021. Is homophily a necessity for graph neural networks? *arXiv preprint arXiv:2106.06134*.

Navarin, N.; Oneto, L.; and Donini, M. 2020. Learning deep fair graph neural networks.

Paige, C. C.; and Saunders, M. A. 1981. Towards a generalized singular value decomposition. *SIAM Journal on Numerical Analysis*, 18(3): 398–405.

Petersen, F.; Mukherjee, D.; Sun, Y.; and Yurochkin, M. 2021. Post-processing for individual fairness. *Advances in Neural Information Processing Systems*, 34: 25944–25955.

Saxena, A.; Fletcher, G.; and Pechenizkiy, M. 2022. HM-EIICT: Fairness-aware link prediction in complex networks using community information. *Journal of Combinatorial Optimization*, 44(4): 2853–2870.

Shchur, O.; Mumme, M.; Bojchevski, A.; and Günnemann, S. 2018. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*.

Srinivasan, B.; and Ribeiro, B. 2019. On the equivalence between positional node embeddings and structural graph representations. *arXiv preprint arXiv:1910.00452*.

Tang, J.; Zhang, J.; Yao, L.; Li, J.; Zhang, L.; and Su, Z. 2008. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 990–998.

Tang, L.; and Liu, H. 2009. Relational learning via latent social dimensions. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 817–826.

Tsioutsiouliklis, S.; Pitoura, E.; Tsaparas, P.; Kleftakis, I.; and Mamoulis, N. 2021. Fairness-aware pagerank. In *Proceedings of the Web Conference 2021*, 3815–3826.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Wang, D.; Lin, J.; Cui, P.; Jia, Q.; Wang, Z.; Fang, Y.; Yu, Q.; Zhou, J.; Yang, S.; and Qi, Y. 2019. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE International Conference on Data Mining (ICDM)*, 598–607. IEEE.

Wang, N.; Lin, L.; Li, J.; and Wang, H. 2022. Unbiased graph embedding with biased graph observations. In *Proceedings of the ACM Web Conference 2022*, 1423–1433.

Wasilewski, J.; and Hurley, N. 2016. Incorporating diversity in a learning to rank recommender system. In *The twenty-ninth international flairs conference*.

Xu, B.; Cui, Y.; Sun, Z.; Deng, L.; and Zheng, K. 2021. Fair Representation Learning in Knowledge-aware Recommendation. In *2021 IEEE International Conference on Big Knowledge (ICBK)*, 385–392. IEEE.

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

Zemel, R.; Wu, Y.; Swersky, K.; Pitassi, T.; and Dwork, C. 2013. Learning fair representations. In *International conference on machine learning*, 325–333. PMLR.