FAIRNESS IN SOCIAL NETWORK ANALYSIS: MEASURES AND ALGORITHMS

By

Farzan Masrour Shalmani

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Computer Science – Doctor of Philosophy

2021

ABSTRACT

FAIRNESS IN SOCIAL NETWORK ANALYSIS: MEASURES AND ALGORITHMS

By

Farzan Masrour Shalmani

The use of machine learning in human subject-related tasks has resulted in growing concerns about the inherent biases within such automated decision-making algorithms. In response to these concerns, we are witnessing a growing body of literature that focuses on designing fairness-aware machine learning algorithms. However, current fairness research is mostly limited to non-relational, independent and identically-distributed (i.i.d) data. To overcome this limitation, this thesis aims to develop fairness measures and algorithms for analyzing social networks, which is an important class of relational data. In particular, this work investigates the challenges of ensuring fairness in link prediction, node classification, and network sampling, which are three important network analysis tasks. First, we develop a novel fairness-aware link prediction framework that combines adversarial network representation learning with supervised link prediction based on network modularity measure. We show that this approach promotes more diverse links and addresses the filter bubble problem in social networks. Second, we investigate the node classification problem from a fairness perspective. We introduce a novel yet intuitive measure known as fairness perception and provide an axiomatic approach to analyze its properties. A fairness-aware classification algorithm is developed to balance the trade-off between maximizing accuracy and minimizing the perception of bias in the classification decisions. Using a graph-theoretic framework, we present a theoretical bound on the gap between the true positive rates for different groups of individuals when fairness perception is maximized. Finally, we investigate the network sampling problem from a fairness perspective. Specifically, we propose a novel fairness-aware network sampling framework that combines the structural preservability and group representativity objectives into a unified structure. We also present a fair greedy sampling algorithm with bounded approximation guarantees.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

CHAPTER 1

INTRODUCTION

The transformative impact of machine learning on our daily lives is undeniable. Integration of machine learning algorithms into decision making in various sectors such as healthcare [175, 18], banking [103], and criminal justice system [15, 62] are becoming increasingly popular. Nevertheless, it has also resulted in growing concerns regarding the potential implicit bias in the decision outcomes of these algorithms [11]. In particular, the output of these algorithms may discriminate against particular subgroups in the population, as identified by their protected attributes such as gender, race, or sexual orientation. For instance, a previous study [11] has suggested that the risk assessment software used in courtrooms in Florida discriminates against African Americans. The study found that black defendants were nearly twice more likely to be misclassified as higher risk for recidivism than their white counterparts. In addition, discrimination in machine learning decisions has been reported in other situations like hiring [42], credit scoring [2], and medical diagnosis [101].

There have been concerted efforts to address these concerns in recent years. However, despite the increasing body of literature on quantifying fairness and developing fair machine learning algorithms, existing studies are mostly limited to non-relational data, which assume that the data instances are independent identically distributed (i.i.d). Relational data violates this assumption of independence as the individual data instances can have direct relations with one another [51]. For instance, there is a correlation among the underlying content or topics of two papers in a citation network if one cites the other.

An important class of relational data is that of network data. Networks are powerful tools for modeling interactions between entities in a complex system. Examples include the Internet as a physical network of interconnected computing devices, social media platforms such as Facebook for human communications, and gene regulatory networks in biochemical systems. Analyzing the properties of these networks may provide useful insights into the underlying mechanism governing the behavior of the complex system. In particular, substantial progress has been made to address fundamental questions such as: How are the links established in a network? How do communities formed and sustained over time? To provide answers to these and many other network related questions, innovative algorithms have been developed to mine the rich and rapidly growing repositories of network data.

This thesis focuses on developing fairness measures and algorithms for social networks. The key research question of this thesis is as follows: *How to develop a fairness-aware network analysis framework?* In other words, how to detect and mitigate bias in algorithmic decisions in network data? In the rest of this chapter, I will first describe this challenge in more details and explain why it is an important problem. I will then discuss the main contributions of this work and provide an outline of the rest of the thesis.

1.1 Fairness in Network Analysis

Social network data have proliferated over the past decade. By 2016, 80% of all mobile owners in the US have switched to smartphones, which is a 38% increase compared to just five years before[1]. This means more online users, more online purchases, and more social networking apps. Many of these apps employ machine learning algorithms, including network-based models, to provide a variety of services to their customers. However, recent studies [10, 99] have shown that these algorithms can be prone to biases. For example, Ali et al. [10] showed that the ad targeting algorithms behind online digital advertising platforms such as Facebook have significant bias in their advertisement delivery along gender and racial lines.

In order to address biases and discriminatory decisions in machine learning, the following questions need to be answered: (1) How to objectively quantify whether an algorithmic decision is fair/unfair? (2) How to design algorithms that will mitigate the unwarranted biases and prevent discrimination against individuals or underrepresented groups? For i.i.d. data, researchers have proposed several definitions of fairness that generally compare an algorithm's performance on

[1] https://www.freshbooks.com/blog/smartphone-revolution-small-businesses

different subgroups of the population. This allows us to answer questions such as “Is the minority group more likely to be misclassified than the majority group?" by comparing the disparity in false positive or false negative rates of classification models across different groups. After quantifying the fairness measure, the next step is to design algorithms that account for the fairness measure along with other performance criteria (e.g., model accuracy).

While the same approach can be applied to relational data, answering these questions would require more careful considerations in relational data since the link structure often embeds information about the protected attribute. For example, individuals in a social network have a tendency to form ties with other individuals of the same gender, race, age group, educational background, etc., an effect known as the homophily principle in social network analysis. This may introduce systematic bias in the network data, which in turn, affect the network analysis results For example Karimi et al [83] have shown how homophily can influence the ranking of minorities in real-world networks by restricting their ability to establish links with members of the majority group.

Previous research has also suggested that many fairness metrics are incompatible with each other [92]. To illustrate the difficulty of this problem, consider the well-documented debate on COMPAS, a risk assessment software for predicting recidivism among offenders [173]. ProPublica investigative journalists claimed that the COMPAS algorithm is racially biased due to the significant disparity in false positive and false negative rates between black and white offenders [11]. In response to this analysis [47], Northpointe asserted that their COMPAS software is indeed fair and that ProPublica analysis had ignored the fairness criteria used by their software. Specifically, they showed that the risk score provided by COMPAS did not discriminate against blacks since the likelihood of recidivism predicted by the software is the same regardless of the race of the offenders. Indeed, Kleinberg et al.[92] showed that it would be impossible to satisfy the diverse fairness criteria simultaneously. The main takeaway conclusion here is that the notion of fairness should be application and context dependent. Thus, one should focus on applying the notion of fairness that makes the most sense for the problem domain at hand. This argument can be extended to social network data. Since they violate the i.i.d assumption, one should consider defining fairness measures that take into account the link structure of the network.

1.2 Thesis Contributions

This thesis aims to expand the study of fairness to the mining of network data. The network mining tasks to be investigated in this study include link prediction, node classification, and network sampling. Specifically, I will introduce novel fairness metrics for network data and develop learning algorithms that connsider the trade-off between fairness and utility of the models. The challenges and contributions of this dissertation are summarized in the remainder of this section.

1.2.1 Metrics for Assessing Fairness in Network Data

As previously mentioned, the choice of fairness criteria should depend on the problem at hand. In this dissertation, we introduced three novel fairness metrics depending on the application and context of the network analysis task, i.e., whether it is at node-level, link-level, or subgraph-level. The node-level fairness measure is applicable to node classification tasks while link-level fairness measure can be applied to link prediction problems. The subgraph-level measure, on the other hand, can be utilized for the network sampling problem.

Chapter 3 presents a new link-based network fairness metric, which is an adaptation of the well-known network modularity measure. The modularity of a network is computed by comparing its link density against that of a random network with similar degree distribution. The measure is extended to account for homogeneity of the node pairs that form the links in the network. Specifically, a network with a large proportion of its links between nodes with the same protected attribute value will have a high modularity value while one with more diverse links will have a lower modularity value. This measure can be utilized to promote more heterogeneous (diverse) links, and thus, prevent reinforcing segregation in link-based recommender systems.

Chapter 4 introduces the proposed node-level metric known as network-centric fairness perception. The idea here is that an individual's perception of fairness should depend on his/her expectation of the decision outcome. For example, if the individual expects a positive outcome and indeed receives a favorable decision from the algorithm, the individual will likely perceive the decision to be fair. However, if a positive decision is expected yet a negative outcome is received, the individual will likely perceive the decision as unfair. The question is, how do we determine the expected outcome of an individual? In social comparison theory, the expected outcome depends on members of the reference group to whom the individual identified with. For network data, the reference group that helps shape an individual's expectation is defined by the node's local neighborhood. For example, a node expects a positive outcome if all of its neighbors also receive a positive decision outcome. The outcomes of neighboring nodes thus help to shape the expected outcome, which in turn, can be used to compute the fairness perception of an individual node.

Chapter 6 introduces the proposed subgraph-level fairness metric known as max-min subgraph fairness. It is based on the idea that fairness measure on a subgraph should be measured in terms of its worst-case quality measure across all subgroups of the protected attribute. The metric is inspired by the idea proposed in [118] for classification of i.i.d. data, which evaluates group fairness as a minimax problem instead of the standard approach of measuring disparity in the outcomes for different groups. In this thesis, the proposed max-min subgraph fairness measure is used for the network sampling problem.

1.2.2 Fairness-Aware Network Mining Tasks

After offering network fairness measures, the next challenge is to introduce fairness into network analysis tasks without significantly sacrificing accuracy. In this section we discuss the main findings concerning this challenge.

1.2.2.1 Fairness-Aware Link Prediction

Homophily, which is the tendency of individuals to form relations with others similar to them, is an essential characteristic of many social networks. Homophily can give rise to network community structure with potential discriminative consequences in some situations. For example, consider professional networking sites. Certain professions, such as software engineering, are male-dominated and intensely segregated. This phenomenon, which is reflected in the network's community structure, can adversely effected the employment opportunities of talented female engineers and other minorities. Homophily can be quantified using the well-known network modularity (or assortative mixing) measure.

As we discussed above, due to the homophily principle, current link prediction algorithms are susceptible to promoting links that may increase the network's segregation. To mitigate this problem, we proposed two algorithmic solutions in Chapter 3. The first algorithm is a greedy postprocessing approach that utilizes the proposed fairness criteria based on modularity measure. This algorithm can be applied to the output of any link prediction algorithm, and a hyper-parameter can specify the importance of accuracy versus fairness measure. The second algorithm is a novel framework that combines adversarial network representation learning with supervised link prediction. The architecture of the algorithm consists of three components: A generator that learns node representations, a discriminator that tries to predict link types, and a link prediction component for new link inference. Experimental results on several real-world datasets showed the effectiveness of the proposed methods in reducing the predicted network's modularity without degrading prediction accuracy significantly.

1.2.2.2 Fairness-Aware Node Classification

The network-centric fairness perception measure described previously can be used as a criterion for assessing fairness in node classification decisions. In Chapter 4, we presented a case study on a peer-review network to show how fairness perception can be exploited to mislead individuals into perceiving unfair algorithmic decisions as fair. In Chapter 5, we investigated this issue theoretically. The main theoretical result of this work is finding an upper bound on true positive rate disparity, i.e., the gap between true positive rates for two distinct groups of the protected attribute, for a classifier that maximizes fairness perception. This upper bound is a function of the network structure and the distribution of nodes belonging to the protected groups in the network. When the upper bound is small, we showed that maximizing fairness perception has a linear solution. In this case, fairness perception will not only ensure that individuals are satisfied with the classification decision but also guarantees fairness in terms of true positive rate parity. When the upper bound is large, we proposed a novel multi-objective optimization algorithm to achieve a compromise between maximizing fairness perception and minimizing the true positive rate disparity.

1.2.2.3 Fairness-Aware Network Sampling

Network sampling aims to achieve the following two objectives: (1) to preserve specific properties of the original network. We refer to this objective as structural preservability. (2) to obtain a representative subset of nodes. We will refer to this objective as group representativity. A fair sample can be defined as a sample that satisfies both structural preservability and group representativity. Chapter 6 shows how the proposed max-min subgraph fairness measure can be used as a unifying framework that combines structural preservability and group representativity.

A greedy algorithm was proposed to generate a representative and preservative subgraph given a target set of nodes. The algorithm utilizes a preservability measure based on the harmonic mean node centrality measure. An approximation guarantee for the output of the proposed greedy algorithm based on submodularity and curvature ratios is also presented. The proposed sampling strategy can be also employed during the mini-batch training process of a graph convolutional network (GCN) to generate an embedding of similar quality for each subgroup of the population.

1.3 Outline

The remainder of this thesis will be organized as follows: Chapter 2 discusses the related work and provide a formal overview of the material related to this thesis. In Chapter 3, we move forward to the main contribution of this thesis and discuss the filter bubble problem from a fairness perspective. Chapters 4 and 5 are related to the fairness perception problem. While Chapter 4 mainly focused on the introduction of the notion of fairness perception and its properties. Chapter 5 presents the proposed algorithm to mitigate the perception bias. Chapter 6 investigates the fairness-aware network sampling problem. Finally, Chapter 7 discusses the plans for future research directions on network analysis and fairness-aware machine learning.

1.4 Bibliographic Notes

The materials in this thesis are based on the following publications:

1. **Mitigating Perception of Bias in Peer-Review Decision Making** F Masrour, P Tan, A Esfahanian, in Proceedings of IEEE International Conference on Data Mining (**Under Review**).

2. **Algorithmic Fairness Perception: A Network-Centric Perspective** F Masrour, P Tan, A Esfahanian, in Proceedings of IEEE International Conference on Data Mining (**ICDM**), 2020.

3. **Bursting the Filter Bubble: Fairness-Aware Network Link Prediction.** F Masrour, T Wilson, H Yan, P Tan, A Esfahanian, in Association for the Advancement of Artificial Intelligence(**AAAI**), 2020.

During my research at Michigan State University, I collaborated with several colleagues on multiple deep learning and optimization research projects with applications in network analysis. The following are my additional publications related to this research:

1. **OPTANE: An OPtimal Transport Algorithm for NEtwork Alignment.** F Masrour, P Tan, A Esfahanian, in International conference on Advances in Social Networks Analysis and Mining(ASONAM), 2019.

2. **You have been CAUTE! Early Detection of Compromised Accounts on Social Media.** C VanDam, F Masrour, P Tan, T Wilson, in International conference on Advances in Social Networks Analysis and Mining(ASONAM), 2019.

3. **Attributed Network Representation Learning Approaches for Link Prediction.** F Masrour, P Tan, A Esfahanian, C VanDam in International conference on Advances in Social Networks Analysis and Mining(ASONAM), 2018.

4. **Network Completion with Node Similarity:A Matrix Completion Approach with Provable Guarantees.** F Masrour, R Forsati, I Barjesteh, in International conference on Advances in Social Networks Analysis and Mining(ASONAM), 2015.

5. **PushTrust: Efficient Recommendation by Leveraging Trust and Distrust Relations.** R Forsati,I Barjesteh ,F Masrour,The ACM Series on Recommender Systems(RecSys), 2015.

6. **Cold-Start Item and User Recommendation with Decoupled Completion and Transduction.** I Barjesteh, R Forsati, F Masrour, The ACM Series on Recommender Systems(RecSys), 2015.

CHAPTER 2

LITERATURE REVIEW

This chapter has two goals. First, a formal overview of the material related to this work. Second, a comprehensive and critical synthesis of the state of art works related to network mining. In particular, we first introduce a formal definition of a network. Then we discuss the main network mining tasks.

2.1 Networks

In this section, we first introduce basic concepts and definitions in the field. Then we discuss some quantitative properties and theoretical ideas of graph theory necessary for the discussions in the following chapters.

2.1.1 Basic Concepts and Definitions

A network is an abstract representation of real-world concepts focusing on entities and the interactions between them. In math literature, a graph is a structure consisting of a set of objects and a set of object pairs which indicates the relation between the objects. There is no difference between a graph and a network, and in this work, we use these terms interchangeably. An information network is a particular category of networks such that entities refer to a form of data. Example of information network includes World Wide Web, online social networks and academic citation networks. In this thesis, we mostly focus on attributed information networks, which can be defined formally as follow:

Definition 1. *An **attributed network** is a directed graph $G =< V, E, F >$, where V is the set of nodes(vertices) in G, E is the set of links(edge) between nodes in G where each element in this set is a pair of nodes in V, and F is the corresponding node attributes matrix such that each row in F represent the feature vector of nodes in V.*

A undirected attributed network is a special case of attributed network, where if $(v_i, v_j) \in E$ then also $(v_j, v_i) \in E$. An alternative way to represent a network is by using *adjacency matrix*. The adjacency matrix $\mathbf{A}$ of network G is a $n \times n$, $n = |V|$, matrix with elements $\mathbf{A}_{ij}$ such that

$$\mathbf{A}_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

We also define *immediate neighbors* of node v_i as follow:

$$N(v_i) = \{v_j | \, (v_i, v_j) \in \mathcal{E}\} \tag{2.2}$$

Another notation we will use in the rest of this work is degree vector $d \in \mathbb{R}^n$ for $n = |V|$. where

$$d_i = |N(v_i)| = \sum_j \mathbf{A}_{ij} \tag{2.3}$$

For directed networks the above equation defines the *out degree* vector and for *indegree vector* we have

$$d_i = \sum_j \mathbf{A}_{ji} \tag{2.4}$$

A *walk* on a network is a sequence of nodes $v_1, v_2 \ldots v_k$ such that for any node v_i in the sequence $(v_i, v_{i+1}) \in E$. A *trial* in the network is a walk in which all links are distinct. A *path* is a trail which all the nodes in the walk are distinct. A *cycle* is a non-empty trail such that the only repeated nodes are the first and last nodes. Distance of node v_i from node v_j is represents by $dist(v_i, v_j)$, and it is equal to the length of the shortest path starting from node v_i and end in node v_j.

A network is *connected* if there exist a path that connects every pair of nodes in the network. A network is said to be *aperiodic* if there is no integer $k > 1$ that divides the length of every cycle of the network.

2.1.2 Random Walk

A *random walk* is a sequence of nodes on a network generated by traversing network randomly. In particular, if $\mathbf{P}$ represent the *transition matrix* such that $\mathbf{P}_{ij}$ is the probability of reaching node v_i

from node v_j such that for all v_i we have

$$\sum_j \mathbf{P}_{ij} = 1$$

Then a random walk is the process of starting from a node in the network and traversing the network according to the transition matrix $\mathbf{P}$. One can define $\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$ where $\mathbf{D}$ is the degree diagonal matrix, $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{\mathbf{ij}}$.

Let $p(t) \in \mathbb{R}^n$ represents the probability vector(distribution) of a random walk on a given network where $p_i(t)$ is the probability of visiting node v_i at step t of random walk. $p(t)$ can be expressed by the following recursive equation:

$$p(t+1) = \mathbf{P}^\top p(t) = (\mathbf{D}^{-1}\mathbf{A})^\top p(t) \tag{2.5}$$

And the *stationary distribution* can be defined as follow

$$\pi = \lim_{t\to\infty} p(t) \tag{2.6}$$

By using *fundamental theorem of markov chains* [38], it can be shown that a random walk will converge to a unique stationary distribution regardless of the choice of starting node if the network is connected and aperiodic.

2.1.3 Network Measures and Metrics

In order to gain a better understanding of a network and capture particular feature of it one can measure some useful quantities. In the following section we will introduce some useful measures which will be apply in developing baseline algorithms and analysing networks in following chapters.

2.1.3.1 Network Centrality

When we are dealing with networks an important research question is "Which nodes in the network are important or central one?" There is no unique answer to this question and many centrality measures have been proposed [130]. Here we describe some of the most important centrality measures.

The simplest possible definition of importance of nodes in a network is degree size.

Definition 2. *Degree Centrality C_d for a node v_i in a network is $C_d(v_i) = d_i$*

For directed network one can define degree centrality base on indegree, outdegree, or summation of both. The fact that degree centrality is a simple concept does not undermine it's importance. For instance, in social network users with high number of connections have more influence or more prestige than those with less connections.

While number of connections is an important measure of centrality. A generalized version of it is *eigenvector centrality* which incorporate the importance of the neighbors of a node.

Definition 3. *Eigenvector Centrality C_e for a node v_i in a network is*

$$\begin{aligned} C_e(v_i) &= \frac{1}{\lambda}\sum_j \mathbf{A}_{ij} C_e(v_j) \\ C_e &= \frac{1}{\lambda}\mathbf{A}^\top C_e \end{aligned} \tag{2.7}$$

where λ is some fixed constant.

If the graph be connected, by using *Perron-Frobenius Theorem* [184] we can show the eigenvector corresponding to the largest eigenvalue of adjacency matrix is equal to C_e.

One challenge with the eigenvector centrality is the fact that a high centrality node with high degree pass high centrality to all its neighbors. However not everyone known by a well known person is well known. To address this problem one solution is PageRank centrality.

Definition 4. *PageRank centrality C_p for a node v_i in a network is*

$$\begin{aligned} C_p(v_i) &= \alpha \sum_j \mathbf{A}_{ij} \frac{C_p(v_j)}{d_j} + \beta \\ C_p &= \alpha \mathbf{A}^\top \mathbf{D}^{-1} C_p + \beta \mathbf{1} \end{aligned} \tag{2.8}$$

Where **1** is a vector of all one. α and β are user specify scalar. In practice $\alpha < \lambda$ where λ is the largest eigenvalue of the adjacency matrix of the network. For directed network d_j in the above definition will be replaced by out degree and it should be nonzero.

A completely different path to measure a network nodes importance is *closeness centrality*. This centrality measures the mean distance from a node to other nodes in the network. However, this definition results in low value to more central nodes which is opposite of the behavior of other centrality measures we defined above. A commonly approach is to consider the inverse of the mean distance:

Definition 5. *Closeness centrality C_c for a node v_i in a network is*

$$C_c(v_i) = \frac{n}{\sum_j dist(v_i, v_j)} \tag{2.9}$$

Another centrality measure base on the pairwise distance is *betweenness centrality*. It is base on the number of times a node lies on paths between other nodes.

Definition 6. *Betweenness centrality C_b for a node v_i in a network is*

$$C_b(v_i) = \sum_{j \neq i, k \neq i} \frac{n^i_{jk}}{n_{jk}} \tag{2.10}$$

where n^i_{jk} is the number of shortest path from node v_j to v_k that pass thought node v_i. n_{jk} is total number of shortest path from v_j to v_k.

2.1.3.2 Homophily and Assortative Mixing

Homophily [123, 100], which is the tendency of individuals to form relations with others similar to them, is an important characteristic of many social networks. Such relationship can be quantified using the well-known network modularity (or assortative mixing) measure [134, 133]. The measure, which was originally developed for community detection in networks, is based on the idea that a random graph is not expected to contain any clustering structure. Any community structure in a given network can thus be validated by comparing its link density against its expected density if the link structure of the network is completely random. The modularity measure is defined as follows [134]:

$$Q = \frac{1}{2m} \sum_{ij} \Big(A_{ij} - \frac{d_i d_j}{2m} \Big) \delta(c_i, c_j) \tag{2.11}$$

where $\delta(c_i, c_j)$ is the Kronecker delta function, c_i is the community of node i, and m is total number of links. Intuitively, a network is said to be assortative if a significant portion of its links are between nodes that belong to the same community.

2.2 Network Mining Tasks

In this section we will provide a comprehensive study of state of art works related to network mining. We categorize the network mining related tasks into three groups: (1) Nodes oriented tasks, (2) link oriented tasks, and (3) sampling and summarizing tasks.

2.2.1 Node Oriented Tasks

The first category of network mining are the tasks which focus on nodes and includes classification, clustering and community detection, anomaly and spam detection, and ranking. In this section, we review these tasks.

2.2.1.1 Classification

Node classification is a classic problem in network mining with applications ranging from online social network and e-commerce to computational biology. In node classification problem we have access to labels of a subset of nodes, and the task is to infer labels of the remaining nodes in the network. Different from traditional classification researches, because of none-IID nature of the network data, node classification has received considerable attention. In network classification link information can be combined by using label propagation [193, 168], logistic regression[115], and graph regularization [192].

With recent progress in deep learning and neural network study, graph neural network also gained a great deal of attention among researchers which results in novel deep models for solving node classification. Deepwalk [144] is the first well known work in this group. Inspired by natural language processing the DeepWalk method takes the output of multiple random walks on network to learn a latent representations of nodes. Kipf and Welling[89] introduced graph convolutional networks (GCNs) which is a variant of convolutional neural networks operate directly on graphs. Matthias et al, [55] introduced SplineCNN a novel convolution operator based on B-splines, that makes the computation time independent from the kernel size. Veličković, et al. [164] present graph attention networks (GATs), leveraging masked self-attentional layers to address the shortcomings of prior methods based on graph convolutions or their approximations.

2.2.1.2 Clustering and Community Detection

The cluster or community structure property of network is an important feature of real networks, which is based on the assumption that the network structure is consists of modules commonly called communities or clusters and the goal of community detection or clustering algorithm is to partition nodes into these components. There are a variety of community detection algorithms which can roughly divided into two categories: Structural clustering and attributed network clustering. The structural clustering methods only consider the network topology. Methods based on min-cut problem [63] cut the network into several partitions and assume these partitions represent communities. The size of the cut is the number of edges that are being cut. Methods based on graph Laplacian [132] assume that similar nodes should be mapped closer. Quasi-clique detection methods are based on extracting dense subgraph structures [3, 142]. Methods based on matrix factorization [179] factorize the adjacency matrix into node embeddings. In contrast, Attributed graph clustering takes into account both network topology and node side information [181, 25, 180].

More recently, there are methods developed for network clustering and community detection using neural network models. Zhang et al., [190] proposed an adaptive graph convolution method for network clustering that utilise high-order graph convolution to detect cluster structure. Pan, et al.,[140] proposed adversarially regularized graph autoencoder (ARGA) for graph clustering and other network mining tasks. Kipf et al., [90] proposed VGAE a framework for unsupervised learning on graph-structured data based on the variational auto-encoder(VAE).

2.2.1.3 Anomaly Detection

An anomaly is defined as an unusual activity exhibiting a different behavior than others present in the same structure [84]. In network setting one application of anomaly detection is network intrusion detection, where have computers sending packets to each other, and the goal is to find misbehaved nodes. Another application is detecting fake or compromised account in social network. Network anomaly detection can be defined as the task of finding the network objects (nodes/edges/substructures) that are rare and that differ significantly from the majority of the reference objects in the network [8].

Similar to clustering and community detection, network anomaly detection task can be divided into (1) plain network and (2) attributed network anomaly detection. For the plain network a common approach for detecting anomalous network objects is feature extraction[7, 71, 82, 66]. Basically, these approaches use the network structure to compute measures associated with the nodes, edges, subgraphs and communities and use these measures to find anomalous entities. For example, [7] proposed the oddball algorithm for finding anomalous nodes by extracting rules (power laws) in density, weights, ranks and eigenvalues which govern the ego network of social network users.

2.2.2 Link Oriented Tasks

The second category of network mining are the tasks which focus on links and relation and includes link prediction and, Network alignment. In this section, we review these tasks.

2.2.2.1 Link prediction

Link prediction is a well studied problem in network studies [111, 9, 125, 119]. Given a network $G = (V, E)$, let $E' \subset E$ denote a set of observed links and $E - E'$ denote a set of unobserved links in the network, where $|E'| < |E|$. The goal of link prediction is to accurately predict the unobserved links from the partially observed network (V, E'). One of the major obstacle of link prediction problem is class imbalance. In real network the $|E|$ is significantly smaller than number non existing links.

Various algorithms for link prediction have been developed over the past two decades. This include simple algorithms that consider the pairwise similarities between nodes, where the similarity is defined according to the network topology or social science theories [171]. This group of algorithms includes as neighborhood based measures such as common neighbors [44] or Jaccard coefficient which calculates the portion of common neighbors for a given nodes pair. Another two well known algorithms in this category are Adamic/Adar(Ad-Ad), a similar measure that assigns less weight to more connected common neighbors, and preferential attachment (Pr-At)[128], which sets the probability of a connection between two pair of such that it is correlated with the product of the their degrees [131].

Another group of link prediction algorithms is based on global similarity measures. They are defined based on the adjacency or Laplacian matrices of the network. Examples of this category include: Path based methods [98, 137, 111] and random walk based methods [112, 111]. In addition, probabilistic graphical models [34] and matrix factorization [152, 125] methods have also been widely used to address the link prediction problem. Recent years have also witnessed the emergence of deep neural network methods for the link prediction task [107, 108, 109, 160, 188]. Many of these algorithms however suffer from scalability issues due to training time requirement for learning the hidden layers of their architecture [65].

The knowledge graph (KG) is a subgroup of network data structure that represents a semantic network. The node represents a concept, and the edge represents a relation between two entities. KG has gained attention in recent years and it has broad applications from in finance [59], to health [149] and semantic search [177]. Link prediction task can be applied in KG to infer new relations to complete the missing knowledge [172, 45, 191].

2.2.2.2 Network Alignment

Work on network alignment can be generally classified into three approaches: (1) quadratic programming, (2) IsoRank-based, and (3) graphlet-based.

In principle, network alignment can be formulated as an integer quadratic programming problem [13], which unfortunately is a computationally hard problem to solve. Various approximation methods were proposed to overcome this problem. For example, Bayati et al.[13] employed a message passing algorithm based on belief propagation while Klau [91] converted it into a linear programming problem. Other binary relaxation approaches proposed include those by Koutra et al. [95], Zhang and Tong [189] and Vogelstein et al. [165]

IsoRank by Singh et al. [154] is one of the most well-known network alignment algorithm. IsoRank iteratively learns the alignment score between two nodes based on the scores of its neighbors. It can be considered as the pioneer of a class of algorithms in the field. The main idea of the IsoRank algorithm is similar to PageRank algorithm proposed by Page et al. [139]. PageRank defined node similarity recursively in terms of the similarity of the node neighbors and IsoRank utilized this idea by combining two networks using Kronecker product and calculate the alignment index on that network. After IsoRank several improvement version has been presented. Variations of IsoRank include the Liao et al. [110], to reduce computational cost and improve interpret-ability of similarity scores. For instance, Kollias et al. [94] had significant improvements, in terms of computational cost, interpret-ability of similarity scores, and nature of queries. In IsoRankN Liao et al. [110] improved the performance of original IsoRank by utilizing iterative spectral clustering algorithm. Another class of methods is known as GRAAL, which is based on graphlet degree signature vectors. Graphlets are small connected non-isomorphic induced subgraphs of a large network [**?**]. The first algorithm in this class was proposed by Kuchaiev et al. [96], which uses only topological structure to match the networks.

2.2.3 Representation Learning

Network representation learning is an important research problem as the latent features can be used for various network mining tasks such as node classification, community detection, link prediction, etc. The ultimate goal is to construct a set of features for the nodes that are comparable or better than the hand engineered features defined by domain experts [14]. Motivated by the word2vec algorithms [126, 127], using the analogy of representing a network as a document, this has inspired a group of works focusing on representation learning of nodes in networks using random walk based methods [65, 144, 159]. For example, DeepWalk [144] used an approximation of the full softmax called the hierarchical softmax [126] for learning the node embedding whereas the node2vec [65] algorithm uses an alternative method known as negative sampling [127]. This negative sampling strategy was shown to produce better performance compared to the hierarchical softmax approach.

The previous methods consider only the link structure to learn the node representation. Since the nodes of a network contain rich information, Yang et al. [178] showed the equivalence between DeepWalk and matrix factorization and presented a framework that can incorporate node attributes into the network representation learning process. Similarly, Huang et al. [74] proposed the Label-informed Attributed Network Embedding (LANE) framework, which is capable of integrating both the node attribute and label information with the graph structure to enhance the network embedding process. Additionally, Kipf and Welling [89] proposed a semi-supervised convolutional neural networks approach which operates directly on graphs and encodes both the local graph structure as well as node attributes. This work was subsequently extended to an unsupervised learning approach known as GAE in [90]. Closely related to GCN, Hamilton et al. [67] proposed an inductive framework that is capable of learning node embeddings for previously unseen data.

There has been a growing interest in attention models for graphs recently and various techniques have been proposed [164, 32, 54, 68] One of the benefits of attention mechanisms is that they allow for dealing with variable sized inputs and allowing it to “focus on the most relevant parts of the input to make decisions” [164]. Moreover, inspired by Generative Adversarial Network(GAN), there are methods proposed recently for learning stable and robust graph representation. [169, 41]

2.3 Fairness in Machine Learning

Quantifying fairness has been a subject of intense debate among AI and ML researchers in recent years [16, 49, 69, 97]. Previous works are primarily focused on non-relational data and can be classified into two types—individual-level or group-level fairness. Fairness definition at individual level is based on the premise that similar people should be treated similarly. For example, Dwork et al. [49] defined a task-specific metric based on a probabilistic distance measure between individuals via a Lipschitz condition. The metric is used as constraints to optimize a fairness-aware classifier. In contrast, the group-level approach quantifies fairness in terms of statistical measures such as demographic parity, equalized odds [69] or balanced error rate [53] with respect to the protected groups. The measures are typically computed from a confusion matrix [16] and are used to ensure that the average performance do not vary significantly among different groups of a protected attribute.

Let Y be the target variable of interest (true outcome) and X be a set of input features. Conventional supervised learning algorithms are designed to predict the target outcome Y from X by learning a model f such that $\hat{Y} = f(X)$ is the predicted outcome. Existing fairness-aware methods seeks to ensure that the predictions generated by the model will not discriminate against one or more subgroups, defined by a protected attribute X_p such as gender, race, or sexual orientation.

A widely used criterion for assessing fairness is *demographic parity* or *statistical parity*. Demographic parity constrains the output of classification problem to be independent of the protected attribute. Particularly a predictor satisfies demographic parity if there is no correlation between the protected feature and the output. Demographic parity has been used in number of papers [113, 80, 78, 50, 24], which considers the degree of independence between the model output and protected attribute. Assuming both the target outcome and protected attributes are binary-valued, demographic parity seeks to achieve:

$$P(\hat{Y} = 1|X_p) = P(\hat{Y} = 1)$$

In other words, demographic parity wants to equalize the positive decision across different groups of the protected feature.

Another well known fairness criterion is *equalized odds* [69], which seeks to ensure that the predictions are conditionally independent of the protected attribute given the true outcome:

$$P(\hat{Y} = 1 | X_p = 0, Y = y) = P(\hat{Y} = 1 | X_p = 1, Y = y),$$

If we consider $Y = 1$ as advantaged outcome, such as job offer or college admission. Then on can relax the equalized odds to only consider non-discrimination within the advantaged outcome. This relaxation is called *equal opportunity* [69] a predictor $\hat{Y}$ satisfies equal opportunity with respect to protected attribute A and outcome Y, if

$$P(\hat{Y} = 1 | X_p = 0, Y = 1) = P(\hat{Y} = 1 | X_p = 1, Y = 1),$$

In addition to quantifying notion of fairness, there has been growing literature on developing fairness-aware methods. Current methods can be divided into three categories. The first category includes prepossessing algorithms [186, 113, 116] with the motivation that training data is the main cause of bias in machine learning. Zemel et al. [186] introduced an optimization algorithm to map data points into a new space to ensure membership in the protected group is lost. [113] developed a variational autoencoder model for learning node representation that are invariant to protected features while preserving as much of the information as possible. Madras and et al. [116] connected group fairness concept to adversarial concept for learning fair representation.

CHAPTER 3

LINK PREDICTION: FAIRNESS-AWARE PERSPECTIVE

In this chapter, we examine the filter bubble problem from the perspective of algorithm fairness and introduce a dyadic-level fairness criterion based on network modularity measure. We show how the criterion can be utilized as a postprocessing step to generate more heterogeneous links in order to overcome the filter bubble problem. In addition, we also present a novel framework that combines adversarial network representation learning with supervised link prediction to alleviate the filter bubble problem.

3.1 Introduction

Online social networking sites have transformed the way individuals interact and share information with each other. The wealth of social network data available also provide opportunities to mine them for a variety of business applications. For example, businesses can learn about the users' interests, sentiment, and online behavior by analyzing the social network data. The insights gained from such analysis will help businesses to increase engagement with their existing customers or connect with new customers. Despite its importance, recent studies have raised concerns about the potential biases and unintended consequences that may arise from such automated analysis.

For example, link prediction methods [111, 9, 119, 120] are commonly employed by social networking sites to encourage users to expand their social circles. “Suggested for you" on Instagram and “People you may know" on LinkedIn are two example applications of such methods. However, the rise of link prediction systems have led to an effect known as *filter bubble* [141], which is the reinforced segregation and narrowing diversity of information exposed to online users. If left unchecked, the filter bubble may introduce systematic biases in the network data and its subsequent analysis. For instance, Hofstra et al. [72] examined the ethnic and gender diversity of social relationships on Facebook and showed that those who have ample opportunities to befriend other similar users often find themselves in highly segregated networks. This is due to the *homophily* principle [123], which is the tendency of individuals to form social ties with other similar individuals in a network. As current algorithms are designed to promote links between similar users, their suggested links may exacerbate the user segregation problem.

In addition to online social networks, the filter bubble problem is also prevalent in recommender systems, which can be viewed as a link prediction task applied to a bipartite network of users and items. A recent study by Nguyen et al. [135] concluded that recommender systems tend to expose users to “slightly narrowing set of items over time." For example, in movie recommendation, movies from a certain genre may only be recommended to users from a specific gender. By addressing the filter bubble problem in network link prediction, the proposed method can potentially be used to alleviate the filter bubble problem in other types of recommender systems.

This work examines the filter bubble problem for network link prediction from algorithm fairness perspective. Specifically, we consider a link prediction algorithm to be unfair if it is biased towards promoting certain types of links (e.g., those between users with similar gender or other protected attributes). As a motivating example, consider the link prediction task on professional networking sites. Certain professions, such as software engineering, tend to be dominated by men, a fact that is likely to be reflected in the link structure of the professional network. As a result, the links recommended by the site may reinforce this gender-based segregation and primarily recommend links between individuals from the same gender while recommending comparatively fewer inter-gender links. Though such a system may be able to achieve high link prediction accuracy, it may unfairly disadvantage some users. For example, a female software engineer may be treated unfairly as they are seldom recommended to other male software engineers.

Unfair practices due to the decisions generated by automated systems is a problem that has been well-documented in many application domains, including criminal justice, mortgage lending, and university admission. For example, Angwin et al. [11] warned about the potential biases against African Americans in the software used to predict the risk score of defendants who would likely re-offend again while O’Neil [136] cautioned against the manipulative marketing tactics used by for-profit colleges in online advertising that exploit vulnerable populations. These concerns have brought increasing scrutiny into the issue of fairness in machine learning algorithms. Despite their growing research, existing works are primarily focused on independent and identically distributed (*i.i.d*) data, and thus, may not be suitable for link analysis problems. For example, previous works have considered the notion of fairness either at individual [49] or group [69, 53] level. In contrast, this work examines the notion of fairness at a *dyadic-level*, based on the pairwise interactions between users in a social network. Furthermore, previous approaches have considered fairness in terms of the unjust decisions against members of a specific underrepresented (protected) group. Instead, we consider fairness in terms of promoting inter-group connections in a network in order to alleviate the filter bubble problem.

There are four major contributions of this work. First, we empirically assess the influence of protected attributes such as gender on the link structure of a network by measuring the homophily effect on several real-world network datasets. Second, we introduce *modred* as a fairness criterion for network link prediction. The metric is inspired by the well-known modularity measure [134] developed for network community detection. We consider the reduction in modularity measure as a way to determine whether the links predicted by an algorithm may lead to further segregation of the network. We then illustrate how the measure can be incorporated into a greedy algorithm for postprocessing the results of current link prediction algorithms. Finally, we present a novel Fairness-aware LInk Prediction (FLIP) framework that combines adversarial network representation learning with supervised link prediction to mitigate the filter bubble problem.

3.2 Related Work

Link prediction is a well studied problem in network analysis with various algorithms been developed over the past two decades [111, 9, 119]. This includes heuristics methods that consider the pairwise similarities between nodes, where similarity is defined based on the network topology [131, 111] or node features [39, 27]. The main benefit of these methods is their simplicity and the fact that most of these approaches do not required training. Another class of link prediction methods employ machine learning methods, such as those based on probabilistic graphical models [34], matrix factorization [152], and supervised classification [9, 167]. Despite their higher accuracy, these methods often suffer from the class imbalance problem as the number of links in a network is significantly fewer than the number of non-links. Recent years have also witnessed the emergence of deep neural network methods for the link prediction task [108, 109, 160]. These methods have been shown to achieve state of the art performance.

Social networks are increasingly personalizing their content using automated machine learning techniques, which is a concern as the decisions may lead to adverse effects on the users. This is due to the so-called “filter bubble" or “echo chamber" effect [72, 141] in which individuals are increasingly isolated to consuming only information that conform to their own belief system. In online social networks, the effect of filter bubble is exemplified by the recommendation decisions generated using link prediction algorithms. As link prediction algorithms are commonly used to encourage users to expand their networks, this may lead to adverse consequences such as segregation of users [72, 135].

Quantifying fairness has been a subject of intense debate among AI and ML researchers in recent years [16, 49, 69, 97]. Previous works are primarily focused on non-relational data and can be classified into two types—individual-level or group-level fairness. Fairness definition at individual level is based on the premise that similar people should be treated similarly. For example, Dwork et al. [49] defined a task-specific metric based on a probabilistic distance measure between individuals via a Lipschitz condition. The metric is used as constraints to optimize a fairness-aware classifier. In contrast, the group-level approach quantifies fairness in terms of statistical measures such as demographic parity, equalized odds [69] or balanced error rate [53] with respect to the protected groups. The measures are typically computed from a confusion matrix [16] and are used to ensure that the average performance do not vary significantly among different groups of a protected attribute.

In addition, there has been growing literature on developing fairness-aware methods. Current methods can be divided into three categories. The first category includes prepossessing algorithms [186, 113, 116] with the motivation that training data is the main cause of bias in machine learning.

Zemel et al. [186] introduced an optimization algorithm to map data points into a new space to ensure membership in the protected group is lost. [113] developed a variational autoencoder model for learning node representation that are invariant to protected features while preserving as much of the information as possible. Madras and et al. [116] connected group fairness concept to adversarial concept for learning fair representation.

3.3 Fairness for Network Data

We first review the fairness criteria for i.i.d. data. Let Y be the target variable of interest (true outcome) and X be a set of input features. Conventional supervised learning algorithms are designed to predict the target outcome Y from X by learning a model f such that $\hat{Y} = f(X)$ is the predicted outcome. Existing fairness-aware methods seeks to ensure that the predictions generated by the model will not discriminate against one or more subgroups, defined by a protected attribute $X_p \notin X$ such as gender, race, or sexual orientation.

A widely used criterion for assessing fairness is *demographic parity* [113, 80, 78, 50, 24], which considers the degree of independence between the model output and protected attribute. Assuming both the target outcome and protected attributes are binary-valued, demographic parity seeks to achieve:

$$P(\hat{Y} = 1|X_p) = P(\hat{Y} = 1)$$

Another well known fairness criterion is *equalized odds* [69], which seeks to ensure that the predictions are conditionally independent of the protected attribute given the true outcome:

$$P(\hat{Y} = 1|X_p = 0, Y = y) = P(\hat{Y} = 1|X_p = 1, Y = y),$$

If we consider $Y = 1$ as advantaged outcome, a special case for this criterion is known as *equal opportunity* [69], which is defined as follows:

$$P(\hat{Y} = 1|X_p = 0, Y = 1) = P(\hat{Y} = 1|X_p = 1, Y = 1),$$

3.3.1 Dyadic-level Fairness

In this work, we investigate the filter bubble problem from the perspective of algorithm fairness. Specifically, a dyadic-level fairness criterion can be defined based on the protected group membership of individuals participating in the links. Below, we consider two such criteria:

- **Subgroup dyadic-level protection**, where fairness is assessed in terms of how representative each protected subgroup is in the formation of the links. For example, in applications such as link-based recommender systems, the fairness criteria could be to ensure that the recommended links do not favor certain subgroups in the population at the expense of other subgroups.
- **Mixed dyadic-level protection**, where fairness is determined based on homogeneity of the nodes involved in each link. Specifically, a link is considered to be an *intra-group link* if it relates a pair of nodes with the same protected attribute values. Otherwise, it is known as an *inter-group link*. To prevent effects such as filter bubble, inter-group or mixed links should be favored to prevent segregation of the users.

In principle, the subgroup dyadic-level protected can be implemented using existing group-level fairness criteria for i.i.d. data by applying them to the links instead of individual nodes in the network. For mixed dyadic-level protected, we introduce the network modularity measure to be described in the next section.

3.3.2 Network Modularity

Homophily [123, 100], which is the tendency of individuals to form relations with others similar to them, is an important characteristic of many social networks. Such relationship can be quantified using the well-known network modularity (or assortative mixing) measure [134, 133]. The measure, which was originally developed for community detection in networks, is based on the idea that a random graph is not expected to contain any clustering structure. Any community structure in a given network can thus be validated by comparing its link density against its expected density if the link structure of the network is completely random. The modularity measure is defined as follows [134]:

$$Q = \frac{1}{2m}\sum_{ij}\Big(A_{ij} - \frac{d_i d_j}{2m}\Big)\delta(c_i, c_j) \tag{3.1}$$

where A is the adjacency matrix representation of the network, $\delta(c_i, c_j)$ is the Kronecker delta function, c_i is the community of node i, d_i is its corresponding degree and m is total number of links. Intuitively, a network is said to be assortative if a significant portion of its links are between nodes that belong to the same community.

The modularity measure can be used to determine whether a network is unfair in terms of mixed dyadic-level protection by replacing $\delta(c_i, c_j)$ in Equation (3.1) with $\delta(X_{pi}, X_{pj})$, where X_{pi} is the protected attribute value for node i. The Q value is thus influenced by only those pairs of nodes belonging to the same protected class. Values of Q close to one would indicate high unfairness due to the strong alignment between the link structure and the protected attribute while values close to zero indicate high fairness. For numeric-valued protected attributes such as age or income level, it can be modified as follows:

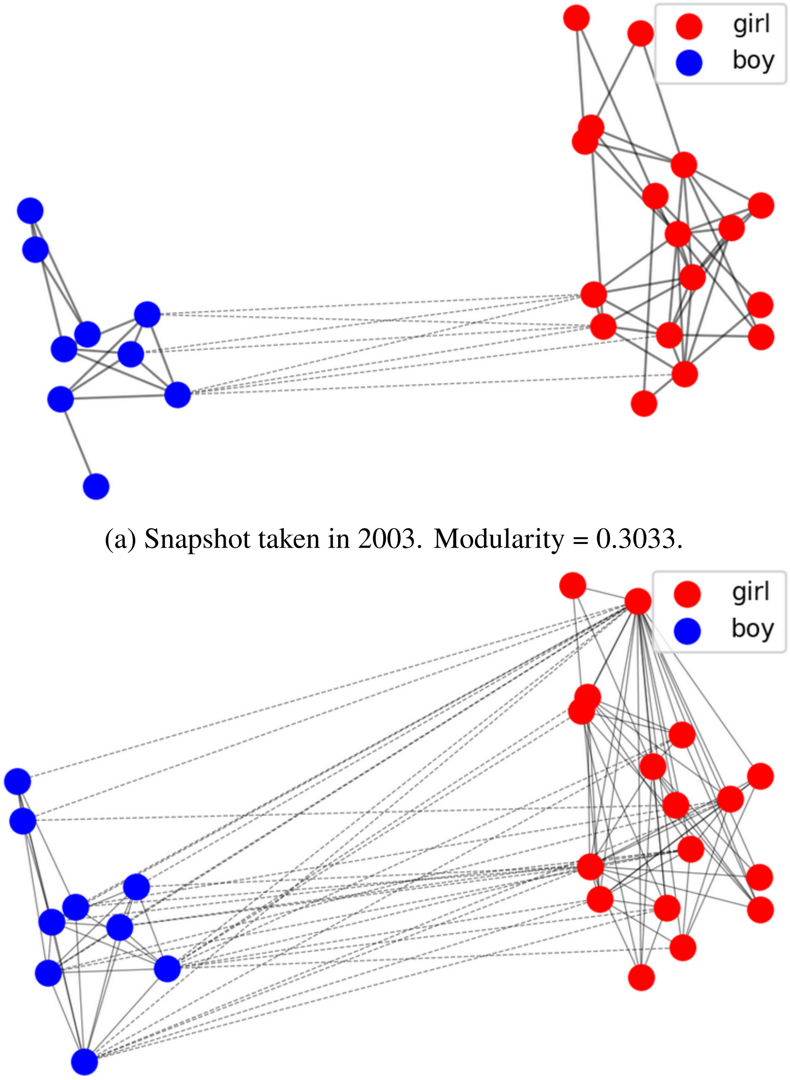
$$r = \frac{\sum_{ij}(A_{ij} - d_i d_j/2m)X_{pi}X_{pj}}{\sum_{kl}(d_k\delta_{kl} - d_k d_l/2m)X_{pk}X_{pl}}$$

where

This is also known as assortativity coefficient of the network.

To illustrate the use of modularity as a measure of unfairness, consider the networks shown in Figure 3.1. The data correspond to friendship relations among freshman at a secondary school in the Netherlands from 2003-2004 [155]. Using gender as protected attribute, the modularity value for the first network shown in Figure 3.1(A) is equal to 0.3033 while the value for the second network is 0.0179. Note that the network with higher modularity has more links between students of the same gender compared to the one with lower value, and thus, is unfair from the perspective of mixed dyadic-level protection.

Our proposed fairness-aware framework evaluates the reduction in the modularity measure to determine whether the modified network obtained from the link prediction results is biased towards



0.50.5

(a) Snapshot taken in 2003. Modularity = 0.3033.

0.50.5

(b) Snapshot taken in 2004. Modularity = 0.0179.

Figure 3.1: Snapshots of friendship relation among students at a Dutch school taken 2003 and 2004 along with their modularity values. The node color represents the student's gender. Darker dashed lines correspond to links between students of different gender while the solid ones correspond to links between students of the same gender.

creating more inter-group or intra-group links. Specifically, we define the following metric:

$$modred = \frac{Q_{\text{ref}} - Q_{\text{pred}}}{Q_{\text{ref}}}, \tag{3.2}$$

where Q_{ref} is the modularity measure of a reference network (e.g., the ground truth network when evaluating link prediction algorithms) and Q_{pred} is the modularity of the predicted network, i.e., the network obtained by augmenting the predicted links to the original network. A positive *modred* value indicates that the link prediction algorithm predicts more inter-group links than the ground truth network while a negative value suggests that the algorithm is predicting more intra-group links than the ground truth network.

3.3.3 Greedy Post-Processing

One approach to promoting fairness in link prediction is to post-process the prediction results. To this end, we propose a greedy algorithm for reducing modularity of the predicted network. It takes as input a set of binarized link predictions, $\{\dot{e}_{xy}\}$ and calculates the change in modularity resulting from flipping the prediction of each node pair. The change in modularity for flipping link $\dot{e}_{xy}$ is:

$$score(\dot{e}_{xy}) = (-1)^{\delta(\dot{e}_{xy})}\frac{1}{2m}\Big(-1+\frac{d_x+d_y-1}{2m}\Big)\delta(c_x,c_y)+ \\ \frac{1}{4m^2}\Big(\sum_{j\in C_x, j\neq y} d_j + \sum_{i\in C_y, i\neq x} d_i\Big) \quad (3.3)$$

where the value of $(-1)^{\delta(\dot{e}_{xy})}$ is -1 if $\dot{e}_{xy}$ is 1 and +1 otherwise, d_x and d_y are the degrees of nodes x and y respectively, c_x and c_y are the protected attribute values of node x and y respectively, $\delta(c_x, c_y)$ is the delta function that returns 1 if x and y are the same protected attribute value and 0 otherwise, and C_x and C_y are the sets of nodes with the same protected attribute value as node x and node y respectively. After computing this score for each predicted link we flip the edges with the lowest scores. This is another approximation since the score for edge should be recomputed after each edge is flipped due to changes in the value of d_x and d_y. The number of link predictions to flip is a hyper-parameter that can be varied depending on the importance of accuracy versus modularity.

3.4 Adversarial Learning for Fair Link Prediction

Consider an attributed network $\mathcal{N} = (V, E, X)$, where V is the set of nodes, $E \subseteq V \times V$ is the set of links, and $X \in \mathbb{R}^{|V|\times d}$ is a matrix corresponding to the set of attribute values associated with the nodes in V. Assume X can be partitioned into submatrices $[X_p, X_u]$, which correspond to the

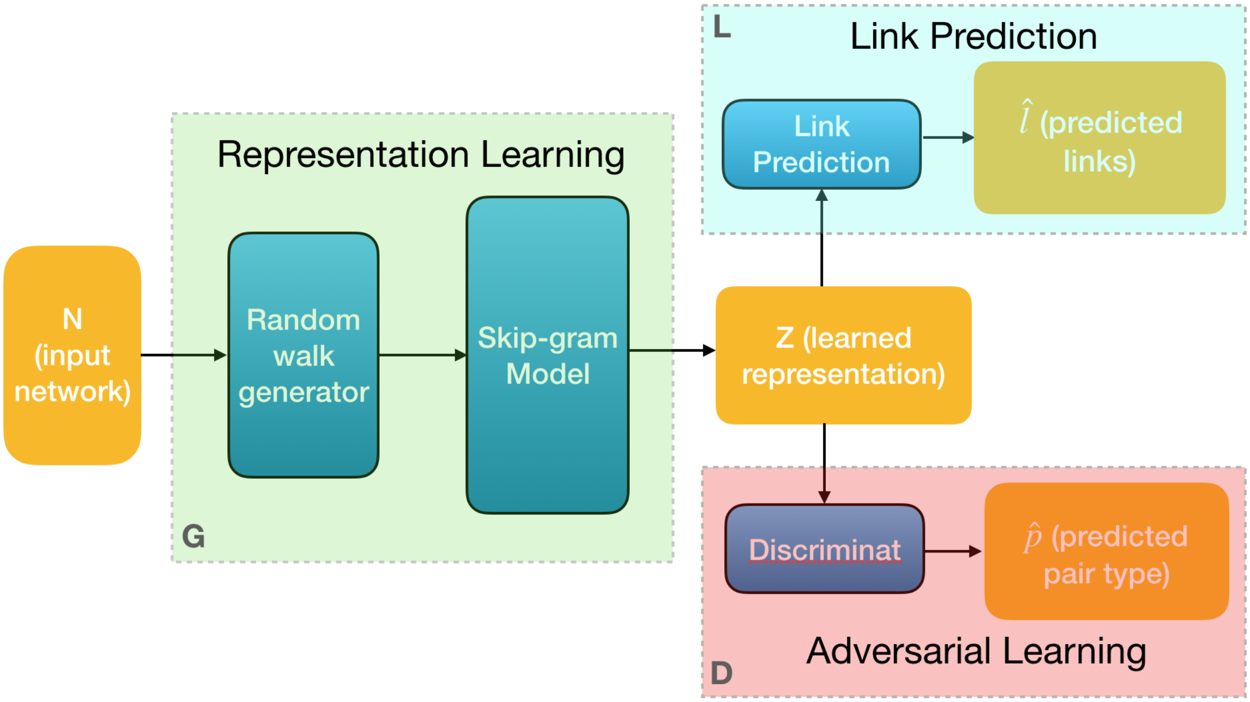


Figure 3.2: FLIP architecture

protected and unprotected features of the nodes. Our goal is to accurately infer new links in the network without being biased against the formation of inter-group links.

Our proposed framework, known as FLIP (Fairness-aware LInk Prediction), employs an adversarial learning approach to ensure that inter-group links are well-represented among the predicted links. The framework consists of the following 3 components, as illustrated in Figure 3.2:

1. A generator, G, that takes the attributed network as input and learns a representation $G(u)$ for each node $u \in V$. We use DeepWalk [144] as the generator, though in principle, the framework can be applied to other network representation learning methods.

2. A discriminator, D, that takes the representations for each pair of nodes produced by the generator as input and attempts to predict if it is an intra-group or inter-group node pair. The discriminator's predicted probability that a pair of nodes has the same protected attribute value is denoted as $D(G(U), G(v))$

3. A link prediction component, L, which tries to infer new links given node representation

learned by the generator. The predicted probability that a link exists between a pair of vertices is $L(G(u), G(v))$.

To understand the rationale behind the framework, note that a good feature representation learned by the generator will enable the link prediction component to infer correctly whether a node pair is connected. If the link structure of the network is biased towards intra-group links, so will the link prediction component as well as the generator. The discriminator plays the role of an adversary who attempts to predict whether a node pair involves nodes from the same group or from different groups. By making the generator and discriminator to work against each other, this would lead to a situation in which the generator produces a feature representation that is good enough for link prediction yet unbiased enough to prevent the discriminator from inferring whether it is an inter-group or intra-group node pair. In networks with homophily property, this will help to discourage the prediction of intra-group links and promotes more inter-group links.

3.4.1 Discriminator

In recent years, adversarial networks have been used to achieve different fairness criteria for independent and identically distributed (i.i.d) data [17, 116]. The shared idea between these methods is an adversarial component that attempts to predict the protected attribute value X_u. A naïve approach to achieving fairness in network data is to follow same path and design an adversarial component that predicts the protected attribute of a node using the following entropy cost function:

$$J^D = -\frac{1}{|V|} \sum_{u \in V} \left[X_{pu} log(\hat{y}_u) + (1 - X_{pu}) \log(1 - \hat{y}_u) \right]$$

Here $\hat{y}_u = D(G(u))$ is the prediction of the discriminator of the binary protected value of node u.

However this will not necessarily result in mixed dyadic level protection because intra-group links may still be favored in a homophilic network. To solve this challenge we propose the following adversarial loss:

$$
\begin{aligned}
J^{D} \;=\; & -\frac{1}{|\mathcal{T}|}\sum_{(u,v)\in\mathcal{T}}\Big[p_{uv} log(\hat{p}_{uv}) \\
& \qquad\qquad +(1-p_{uv})\log(1-\hat{p}_{uv})\Big]
\end{aligned}
\tag{3.4}
$$

where $\mathcal{T} \subseteq V \times V$ is the set of node pairs in the training data, p_{uv} is the actual type of node pair (u, v) with respect to a given protected attribute (i.e. intra-group vs inter-group) and $\hat{p}_{uv}$ is the discriminator's prediction. Instead of inferring the node's protected attribute, the discriminator receives a pair of node representations, which it passes to a two layer fully connected network with leaky ReLU activation to determine the probability that it is an intra-group node pair.

3.4.2 Generator

In contrast to the original GAN framework proposed by [64] where the generator seeks to generate samples of data points that seem real, the generator in our framework tries to learn node representation that will preserve important structural information of the network without implicit usage of the protected attribute information. For the generator, we utilized DeepWalk [144] which is a network representation learning method inspired by the Skip-gram [126] model from natural language processing. DeepWalk consists of two steps: the first step is to extract sequences of nodes from the network by performing a series of truncated random walks starting from each node in the input network. In the second step, the node sequences generated from the random walk process are used to learn the feature representation of each node. This is accomplished as follows. A sliding window of width w scans the generated node sequences to generate all the node pairs (u, v) in which node v appears in the sliding window centered at node u. A fully connected neural network with a single hidden layer predicts the probability of the occurrence of node v given the one hot encoding, $\bar{u}$, of node u. Specifically, the network attempts to predict $p(v|u)$ for each u as follows:

$$
\mathbf{p}(v|u) \simeq \frac{exp(f'(v)^{\top} f(u))}{\sum_{v' \in V} exp(f'(v')^{T} f(u))}
\tag{3.5}
$$

where $f(v) = W\bar{v}$, $f'(u) = Z\bar{u}$, W is the weight matrix between the input and hidden layers of the network, and Z is the weight matrix between the hidden and output layers of the network. The rows of matrix W are the node representations generated by the skip-gram model so we have $G(u) = f(u)$.

The parameters of DeepWalk are trained using the maximum likelihood estimation approach, with the following loss function:

$$\begin{aligned} J^{Skip} = - \sum_{u \in V} \Big[- log(\sum_{v' \in V} exp(f'(v')^{\top} f(u)) \\ + \sum_{v' \in \Omega_W(u)} exp(f'(v)^{\top} f(u)) \Big] \end{aligned}$$

Here $\Omega_W(u)$ represent the set of all nodes that appears in the neighborhood of node u in the given random walk sequence with window size of width w.

3.4.3 Link prediction

This component takes a pair of node embeddings as input to predict whether their nodes should be linked or not. This is accomplished by adding a two-layer link prediction network to the GAN model. During the training phase the link prediction component receives pairs of node embeddings and concatenates them into a feature vector, which is then passed to a two-layer fully connected network with leaky ReLU activation. The output of the network corresponds to the likelihood of a link to exist between the node pair. Here we deployed the standard cross entropy cost function as follows:

$$J^L = -\frac{1}{|\mathcal{T}|} \sum_{(u,v) \in \mathcal{T}} \Big[e_{uv} log(\hat{e}_{uv}) + (1 - e_{uv}) \log(1 - \hat{e}_{uv}) \Big] \tag{3.6}$$

where $\hat{e}_{uv}$ is output of the link prediction component for node pair (u, v) and e_{uv} is the binary ground truth link label.

Putting everything together, the overall loss function for the proposed framework is given as follows:

$$J^G = (1 - \alpha) J^{Skip} - \alpha J^D + \beta J^L \tag{3.7}$$

Table 3.1: The real social network data sets statistics

network	#nodes	#edges	protected feature
Dutch school	26	221	gender
Facebook	1,034	26,749	gender
Google+	4,938	547,923	gender

where β is a hyperparameter. The generator, discriminator, and link prediction network are all trained end to end using Adam [88]. The generator and link prediction network are trained on the same batches but every other batch is used to train the discriminator only so that training alternates between updating the link predictor and generator together one one batch and updating the discriminator on the next batch.

3.5 Experimental Evaluation

This section describes the experiments performed to evaluate the efficacy of our proposed methods to address the filter bubble problem in network link prediction.

3.5.1 Experiment Setup

We first discuss the experimental setup, including data sets, baselines and evaluation metrics used in our experiment.

3.5.1.1 Datasets

We evaluated our methods on three real world social network data sets. Table 6.1 summarizes the main properties of these data sets. The first data set is a Facebook ego network [106], which contains 1,034 nodes, and 26,749 friendship links. The second data set is Google+, which has 4,938 nodes and more than 500,000 links. [106], The third data set is Dutch school network [155], which corresponds to friendship relations among 26 freshmen at a secondary school in the Netherlands. For all three datasets, we use gender as the protected attribute for inferring intra-group and inter-group links.

Table 3.2: Proximity based link prediction algorithms. For each node v, $N(v)$ is the set of its immediate neighbors.

Method	Definition
Jaccard	$\frac{\lvert N(u) \cap N(v) \rvert}{\lvert N(u) \cup N(v) \rvert}$
Adamic/Adar	$\sum_{x \in N(u) \cap N(v)} \frac{1}{log(d_x)}$
Preferential attachment	$d_x d_y$

Table 3.3: Performance comparison of various link prediction algorithms on 3 real-world network datasets.

Method	Dutch school		Facebook		Google+	
	AUC	*modred*	AUC	*modred*	AUC	*modred*
Jac	0.650 +/- 0.000	-0.503 +/- 0.005	0.830 +/- 0.0	-0.149 +/- 0.037	0.793 +/- 0.0	0.093 +/- 0.03
Ad-Ad	0.657 +/- 0.001	-0.376 +/- 0.004	0.84 +/- 0.0	0.222 +/- 0.009	0.869 +/- 0.0	-0.305 +/- 0.001
Pr-At	0.602 +/- 0.002	0.443 +/- 0.002	0.807 +/- 0.0	0.46 +/- 0.001	0.905 +/- 0.0	-2.935 +/- 0.011
DW	0.529 +/- 0.007	-0.047 +/- 0.242	0.951 +/- 0.0	0.089 +/- 0.002	0.771 +/- 0.0	0.166 +/- 0.039
Jac+PEO	0.536 +/- 0.003	0.032 +/- 0.052	0.799 +/- 0.00	-0.557 +/- 0.235	0.750 +/- 0.0	3.47 +/- 0.048
Ad-Ad+PEO	0.527 +/- 0.002	-0.234 +/- 0.052	0.799 +/- 0.00	0.013 +/- 0.06	0.829 +/- 0.0	3.219 +/- 0.013
Pr-At+PEO	0.505 +/- 0.00	0.022 +/- 0.109	0.682 +/- 0.00	-0.21 +/- 0.227	0.858 +/- 0.0	3.854 +/- 0.154
DW+PEO	0.491 +/- 0.005	-0.121 +/- 0.213	0.949 +/- 0.0	0.014 +/- 0.02	0.735 +/- 0.0	3.55 +/- 2.652
Jac+GM	0.657 +/- 0.029	-0.218 +/- 0.084	0.842 +/- 0.00	0.661 +/- 0.05	0.74 +/- 0.0	0.966 +/- 0.04
Ad-Ad+GM	0.653 +/- 0.026	-0.211 +/- 0.088	0.842 +/- 0.00	0.661 +/- 0.05	0.818 +/- 0.0	1.369 +/- 0.038
Pr-At+GM	0.583 +/- 0.044	0.179 +/- 0.165	0.740 +/- 0.00	0.919 +/ 0.06	0.842 +/- 0.00	1.749 +/- 0.096
DW+GM	0.536 +/- 0.056	0.133 +/- 0.091	0.901 +/- 0.00	0.497 +/- 0.062	0.725 +/- 0.03	1.506 +/- 0.591
FLIP	0.658 +/- 0.004	0.359 +/- 0.009	0.86 +/- 0.00	0.348 +/- 0.004	0.857 +/- 0.0	0.207 +/- 0.009

3.5.1.2 Baseline Algorithms

We considered 4 state-of-art link prediction algorithms as baselines. Three of them are well known classical proximity based methods which use neighborhoods structural information. The first baseline is based on the well known Jaccard's (Jac) coefficient similarity metric which is deployed in the context of network link prediction by calculating the portion of common neighbors for a given nodes pair. The second baseline is Adamic/Adar(Ad-Ad), a similar measure that assigns less weight to more connected common neighbors. The third proximity based algorithm is preferential attachment (Pr-At)[128] which sets the probability of a connection between two pair of such that it is correlated with the product of the their degrees [131]. Table 3.2 summarized the formal definition of these algorithms. We also considered the more recent DeepWalk(DW) algorithm [144] which learns d-dimensional feature representations of nodes by simulating uniform random walks and provides latent features for nodes at the first step and then, similar to proposed approach in [65], we construct the edge embedding by applying binary Hadammard product operation to the given node pair and train a logistic regression to do link prediction. For evaluation, we use the settings suggested in the original DW paper for both the DW baseline and the proposed method's skip gram model. These settings are: latent feature dimension (128), length of random walks(80), and number of random walks(10) and window size(10) on all data sets.

We also consider a traditional fairness algorithm based on equalized odds which we use to post-process our 4 baselines. As previously mentioned, imposing an equalized odds constraint on the predictions of a model is a popular way of ensuring fair predictions. For our task, we use a generalized version of equalized odds proposed in [145] to post-process each of the baseline algorithm predictions. To make the generalized equalized odds constraint compatible to network data setting we treat link type, intra-group versus inter-group, as each link's binary protected attribute. We refer to this post processing algorithm as (PEO).

3.5.1.3 Sampling process and training

A big challenge for link prediction algorithms is the sparsity of real world network data. In other words, since the number of existing links are significantly smaller than non-existing links, training a model which is not biased toward negative examples is difficult. Given a graph $N = (V, E, X)$ we generate a training set with equal number of negative and positive examples $< N', E^+, E^- >$. Here N' is the remaining sub-graph after removing all sampled positive links, E^+, and E^- is a set of randomly sampled non-links such that $|E^+| = |E^-|$. Sampling positive links from N is random with the restriction that N' remains connected. For each data set we generating 10 examples of $< N', E^+, E^- >$ by deleting 80% of all links in N. For FLIP and DW we learn node representations by performing random walks on graph N' and train the link prediction using 10% of the generated positive and negative samples. For the other baselines we used all the 30% of E^+ and E^- for calculating the proximity measures. We used remaining 70% for test.

3.5.1.4 Evaluation Metric

We evaluate the quality of link predictions with two metrics, accuracy and the area under the ROC curve (AUC) which represents the trade-off between true and false positives with respect to different thresholds. For the fairness subgroup dyadic-level metric we consider modred measure given in equation 3.2.

3.5.2 Experimental Results

In the following subsection we investigate the general performance of the proposed framework. Due to a lack of space we provide extended experimental results, including parameter sensitivity analysis of FLIP and a FLIP ablation study, in the supplementary file.

3.5.2.1 General Performance

We summarize our results for link prediction in Table 3.3. For FLIP we report the the result for $\alpha = 0.1$, $\beta = 0.2$, epochs=3. For greedy post-processing we chose to the flip the 3% of the predictions that will most reduce the modularity. Based on these results we can make several observations. First, there is generally a trade off between AUC and $modred$ so higher $modred$ scores are only achievable by sacrificing accuracy.

Second, none of the baseline algorithms achieve consistently high $modred$ scores. In particular, equalized odds post-processing provides highly inconsistent gains in $modred$ that are heavily dependant on the data set. It provides significant gains on the Google+ data set, but on the Dutch school data set it provides only moderate gains, and it is a moderate impediment on the Facebook data set. However, greedy post-processing and FLIP always achieve strong $modred$ measures and provide a good balance between AUC and $modred$. This is unsurprising since FLIP and greedy post-processing were the only two techniques specifically designed for promoting fairness in link prediction. This demonstrates the importance of developing algorithms tailored specifically for network data and link prediction.

Third, of all the baselines, preferential attachment generally does the best job of achieving good accuracy and $modred$. One possible explanation for this is that all other baselines make predictions based on the neighborhood structure of nodes. In a network that is homophillic with respect to a protected attribute, nodes with the same protected attribute value are likely to have similar neighborhood structure. Since all of our networks are homophillic with respect to the protected attribute, link prediction methods based on neighborhood structure are more likely to reinforce the existing homophilly and create intra-group links. In contrast, preferential attachment ignores the neighborhood structure of nodes when making predictions so it less affected by pre-existing network homophilly.

3.6 Conclusions

This work presents novel fairness-aware methods to alleviate the filter bubble problem in network link prediction. First, we present a fairness criterion based on network modularity measure to determine whether inter-group links are well-represented in the predicted output of a link prediction algorithm. We then consider two approaches to overcome the filter bubble problem—one based on a greedy postprocessing approach using the *modred* measure while the other based on an adversarial learning framework. Experimental results showed that the proposed methods are promising as they can reduce modularity of the predicted network without degrading prediction accuracy significantly.

CHAPTER 4

FAIRNESS PERCEPTION FROM A NETWORK-CENTRIC PERSPECTIVE

In this chapter, we investigate the issue of algorithmic fairness from a network-centric perspective. Specifically, we introduce a novel yet intuitive function known as fairness perception and provide an axiomatic approach to analyze its properties. Using a peer-review network as a case study, we also examine its utility in terms of assessing the perception of fairness in paper acceptance decisions. We show how the function can be extended to a group fairness metric known as fairness visibility and demonstrate its relationship to demographic parity. We also discuss a potential pitfall of the fairness visibility measure that can be exploited to mislead individuals into perceiving that the algorithmic decisions are fair. We demonstrate how the problem can be alleviated by increasing the local neighborhood size of the fairness perception function.

4.1 Introduction

The influence of machine learning is pervasive across numerous applications, from healthcare and e-commerce to financial and criminal justice systems. Despite its utility, previous studies have shown that the algorithmic decisions may contain unintended biases that discriminate against certain groups of the population [11, 77, 37]. As a result, the challenge of removing biases from the algorithmic decision-making process has gained significant attention in recent years. In particular, various mathematical formulations of fairness have been proposed. For instance, group fairness metrics such as demographic parity and equalized odds have been developed to assess the degree of prejudice against certain protected groups in the population. While each metric has its own merits, many of them are incompatible with each other [33, 58].

The group fairness metrics are designed to determine the level of equity among different groups of individuals who are harmed by or benefited from the algorithmic-driven decisions. However, the consequences of an unfair decision may extend beyond those individuals who are directly impacted by the decision. In fact, they may elicit negative responses from other individuals who identified

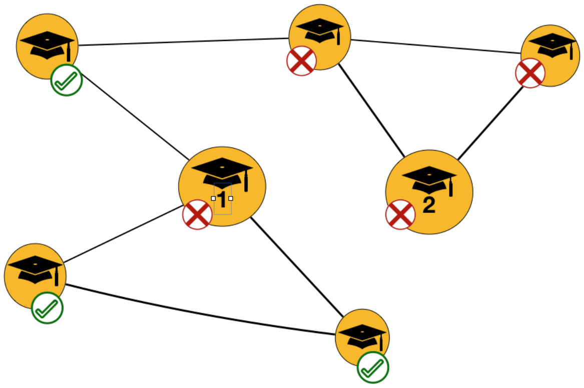


Figure 4.1: An example illustration of fairness perception.

themselves to be in the same group as the affected individuals. For instance, hiring discrimination against a qualified member from an underrepresented group not only affects the well being of that individual, but will also have an adverse effect on other members of the underrepresented group who observed such behavior. This example suggests that fairness assessment must take into consideration the perception of other individuals who may not be directly impacted by the algorithmic decisions [143, 153].

Fairness perception is rooted in the social comparison theory. For instance, equity theory [4] argues that “humans do not base their satisfaction on what they receive but rather what they receive in relation to what they think they should receive". The reaction of an individual to the outcome of a decision process is based on the expectation of the individual and this expectation not only depends on one’s own outcome but also the outcomes of other individuals they are aligned with, which we refer to as the *reference group*. The choice of the reference group is typically influenced by the similarity measure we use, e.g., we may compare ourselves to our co-workers, friends, and family members. By observing the outcomes of other members in our reference group, this will help shape our expectation about what should be considered a fair outcome.

In this paper, we examine the notion of fairness perception from a network analysis perspective. Networks provide a natural way to represent individuals and their connections to other individuals in the same reference group. For instance, Figure 4.1 shows a toy example of a network of students applying for college admission to a prestigious university. In this network, two students are linked if they know each other. Suppose the admission committee of the college has decided to accept 3 of the applicants, denoted as nodes with green check marks, and to reject the other 4 applicants. For brevity, we assume all the students have similar qualifications. Consider the two students labeled as 1 and 2, respectively. Although both applicants were rejected, their expectations for admission and perceptions of fairness are very different. Student 1 has a higher expectation of being admitted compared to student 2 since all of his/her friends were accepted. Thus, the perception of fairness for student 1 is different than that for student 2.

This paper introduces the notion of network-centric fairness perception and illustrates its application to peer review process. Peer evaluation of scientific work has a significant effect on scientific advancement. However, similar to other systems designed by humans, it is potentially biased, favoring certain groups of individuals (e.g., famous researchers from top institutions) [157, 162]. In this study, we show how the proposed function can be used to assess the perception of authors about paper acceptance decisions. An axiomatic approach for analyzing the desirable properties of fairness perception functions is also presented. We then extend our proposed function to a group fairness measure known as fairness visibility and show its relationship to demographic parity under certain mild assumptions. We also describe a potential pitfall of assessing fairness from a local neighborhood perspective. Specifically, it can mislead individuals into thinking that the decision-making process is fair even though the overall decisions are biased toward certain groups of individuals. Finally, we show how to alleviate the problem by expanding the local neighborhood size of the fairness perception function.

4.2 Quantifying Fairness Perception

Let $G =< V, E, X >$ be an attributed network, where V is set of nodes, $E \subseteq V \times V$ is the set of links (edges), and $X \in \mathbb{R}^{|V| \times d}$ is the feature matrix associated with the nodes. We further assume that $X = (X^{(p)}, X^{(u)})$, where $X^{(p)}$ are the protected attributes and $X^{(u)}$ are the unprotected ones. The set of links can also be represented by an adjacency matrix, A, where $A_{ij} = 1$ if a link exists between nodes i and j. Furthermore, we denote $A^k = \prod_{i=1}^{k} A$, where $A^k_{ij} > 0$ if there exists a path of length k between i and j, and 0 otherwise.

We also assume that each node v is associated with a target outcome, $y_v \in \{0, 1\}$. As an example, in the context of peer review network, each node corresponds to a submitted paper and links between papers are established if the two papers share the same authors or have authors who had previously collaborated with each other. The outcome y_v of a given paper v may indicate whether the paper is acceptable or unacceptable based on the average ratings provided by reviewers.

We assume there exists a decision function $h : V \rightarrow \{0, 1\}$ associated with each node in the network. Let $\mathcal{H}$ be the hypothesis space of all decision functions. Our goal is to learn a decision function $h \in \mathcal{H}$ that is consistent with the set of outcomes Y while satisfying some fairness criterion. From the perspective of peer review network, the decision function h may refer to the final decision whether to accept or reject the paper. The true positive and false positive rates of the binary decision function, h, can be computed as follows:

- **True positive rate**, TPR = $\frac{\sum_v y_v h(v)}{\sum_v y_v}$
- **False positive rate**, FPR = $\frac{\sum_v (1-y_v) h(v)}{\sum_v y_v}$

Our goal is to determine how the final decisions are perceived by the individual nodes in the network. Do they feel that the decisions are biased toward nodes that belong to certain groups? To answer this question, we assume each node v is associated with a fairness perception function, $f(v, h)$, given a decision function h. The function provides a local, albeit myopic, view on individual fairness of the nodes in a network.

4.2.1 Axioms for Fairness Perception

We first outline the desirable properties of the fairness perception function, $f(v, h)$, using the following set of axioms. We assume each node $v \in V$ is associated with the following tuple, $(v.X_p, v.X_u, y_v, N(v))$, where $v.X_p$ denotes the value of its protected attribute, $v.X_u$ denotes the value of its other (unprotected) attributes, y_v denotes its target outcome, and $N(v)$ denotes its

δ-neighborhood, which is defined as follows:

$$N(v) = \{u \mid \exists k \leq \delta : A_{uv}^k > 0\}. \tag{4.1}$$

For brevity, we assume $\delta = 1$, unless stated otherwise. Let $G_r = (V_r, E_r, X_r)$ be an ego-network for node r, where $V_r = N(r)$ is the 1-neighborhood of r, $E_r = \{(i, j) \mid i, j \in N(r) \text{ and } (i, j) \in E\}$ and X_r is the feature matrix associated with the attributes of the nodes in V_r. We present a set of axioms on the fairness perception function.

1. **Locality axiom**: If $h(v) = h'(v)$ and $\forall u \in N(v) : h(u) = h'(u)$, where $h, h' \in \mathcal{H}$, then $f(v, h) = f(v, h')$.

2. **Monotonicity axiom**: If $h(v) < h'(v)$ and $\forall u \in N(v) : h(u) = h'(u)$, where $h, h' \in \mathcal{H}$, then $f(v, h) \leq f(v, h')$.

3. **Neighborhood expectation axiom**: If $h(v) = h'(v)$ and $\forall u \in N(v) : h(u) \leq h'(u)$, where $h, h' \in \mathcal{H}$, then $f(v, h) \geq \ f(v, h')$.

4. **Homogeneity axiom**: Let G_u and G_v be the induced sub-graphs of $V_u = N(u) \cup \{u\}$ and $V_v = N(v) \cup \{v\}$, respectively. If G_u and G_v are isomorphic with respect to the decision function h, then $f(u, h) = f(v, h)$.

For the last axiom, we say that a pair of networks, $G_r = (V_r, E_r, X_r)$ and $G_s = (V_s, E_s, X_s)$, are isomorphic with respect to the decision function h if there exists a bijection function $m : V_r \longrightarrow V_s$ such that:

- $\forall u \in V_r : h(u) = h(m(u)), y_u = y_{m(u)}$ and $X_u = X_{m(u)}$.

- $\forall (u_1, u_2) \in E_r : (m(u_1), m(u_2)) \in E_s$

The locality axiom states that the perception of fairness for an individual depends on the decision outcomes for other individuals in its neighborhood. As long as the outcomes for the node and its neighborhood remains unchanged, the fairness perception function should remain the same. The monotonicity axiom suggests that the perception of fairness for an individual never decreases if the decision changes in favor of the individual (assuming the decisions for its neighbors remain unchanged). For example, if a previous decision on the paper was overturned (say from reject to accept), then one should expect the fairness perception to improve (or at least stays the same). In contrast, the neighborhood expectation axiom states that if the number of neighbors with favorable decisions increases, then fairness perception decreases monotonically. This is because, if more individuals in our reference group received favorable decisions, we expect the decision outcome to be favorable for us as well. The increased expectation makes it less likely for us to perceive the decision as fair if our paper is rejected. The fourth axiom ensures consistency of the fairness perception function when applied to different nodes in the network, The axiom states that if two disparate nodes with similar neighborhoods receive the same decision outcomes, their perception of fairness should be the same.

4.2.2 Proposed Network-Centric Fairness Perception

Definition 7 (Network-Centric Fairness Perception)**.** *Given a network $G =< V, E, X >$ and a decision function h, the network-centric fairness perception function is defined as:*

$$f(v, h) = \begin{cases} 1 & \textit{if}\, \mathbb{E}[h(v)] \leq h(v) \\ 0 & \textit{otherwise} \end{cases} \tag{4.2}$$

where $\mathbb{E}[h(v)]$ is the expected value of $h(v)$, which must satisfy the following properties:

1. *If $\forall u \in N(v) : h(u) = h'(u)$, then $\mathbb{E}[h(v)] = \mathbb{E}[h'(v)]$.*

2. *If $\forall u \in N(v) : h(u) \leq h'(u)$, then $\mathbb{E}[h(v)] \leq \mathbb{E}[h'(v)]$.*

3. *Let G_u and G_v be the the induced sub-graphs based on the node sets $V_u = N(u) \cup \{u\}$ and $V_v = N(v) \cup \{v\}$, respectively. If G_u and G_v are isomorphic with respect to the decision function h, then $\mathbb{E}[h(v)] = \mathbb{E}[h(u)]$.*

Our fairness perception function can thus be viewed as a local measure of individual fairness for any given node v in a network. If the decision $h(v)$ is more favorable than expected, then v will perceive the decision as fair. Furthermore, the expected value of the decision outcome, $\mathbb{E}[h(v)]$, depends on the neighborhood of the node v.

Theorem 1. *The network-centric fairness perception function given in Eqn.* (4.2) *satisfies the locality, monotonicity, neighborhood expectation, and homogeneity axioms.*

Proof: The locality and monotonicity properties are proven using the first property. Since $\mathbb{E}[h(v)]$ remains unchanged when $h(u) = h'(u)$ for all the nodes u in the neighborhood $N(v)$, Eqn. (4.2) suggests that $f(v,h)$ depends only on $h(v)$. If $h(v) = h'(v)$, then $f(v,h) = f'(v,h)$, thereby proving that the locality axiom holds. Similarly, if $h(v) < h'(v)$, then $f(v,h) \leq f(v,h')$, which satisfies the monotonicity axiom. For the neighborhood expectation axiom, the second property states that the expected value monotonically decreases when $h(u) \leq h'(u)$ for all the nodes $u \in N(v)$. Since $\mathbb{E}[h'(v)]$ is larger, then nodes that initially satisfy the inequality $\mathbb{E}[h(v)] \leq h(v)$ may no longer do so since $h'(v) = h(v)$. Thus, $f(v,h) \geq f(v,h')$. Finally, we use the the third property to prove the homogeneity axiom. Let G_u and G_v be the induced sub-graphs based on node sets $V_u = N(u) \cup \{u\}$ and $V_v = N(v) \cup \{v\}$, respectively. Since G_u and G_v are isomorphic with respect to the decision function h and $\mathbb{E}[h(u)] = \mathbb{E}[h(v)]$ holds due to the third property, therefore $f(u,h) = f(v,h)$. □

We consider the following **neighborhood peer expectation** approach to compute $\mathbb{E}[h(v)]$:

$$\mathbb{E}[h(v)] = \frac{y_v}{k_1}\Big[\sum_{u \in N(v)} y_u h(u)\Big] + \frac{1-y_v}{k_0}\Big[\sum_{u \in N(v)} (1-y_u) h(u)\Big], \tag{4.3}$$

where $k_0 = \sum_{u \in N(v)} (1-y_u)$, $k_1 = \sum_{u \in N(v)} y_u$, and $y_u \in \{0,1\}$. Note that if the target outcome $y_v = 1$, then $\mathbf{E}[h(v)]$ depends only on the first term (i.e., other nodes u in its neighborhood with $y_u = 1$). On the other hand, if $y_v = 0$, then $\mathbf{E}[h(v)]$ depends only on the second term (i.e., other nodes u in its neighborhood with $y_u = 0$).

Intuitively, the neighborhood peer expectation considers the average decision of all its neighbors with the same target outcome. For example, if y_u denotes whether paper u is acceptable (based on its review ratings) and $h(u)$ is its decision for acceptance, then the expected value of $h(u)$ depends on the average decision for other papers in its neighborhood (e.g., papers co-authored by one of the authors or their collaborators) with the same degree of acceptability. Since the expectation is a monotonically increasing function of $h(u)$ for its neighbors, it can be trivially shown that the neighborhood peer expectation satisfies the first two properties given in Definition 1. The third property holds since the bijection function guarantees that the y and h values for the nodes in the neighborhoods of u and v to be the same. Thus, their expected values, $\mathbb{E}[h(u)]$ and $\mathbb{E}[h(v)]$, will also be the same.

4.3 Fairness Visibility

We now introduce fairness visibility, which extends the fairness perception function to a group fairness measure.

Definition 8 (Fairness Visibility)**.** *Let $V_c = \{u \mid u \in V,\ u.X_p = c\}$, i.e., the set of nodes belonging to the protected attribute group c. The fairness visibility of h for group c is defined as follows:*

$$FV(V_c) = \frac{\sum_{v \in V_c} f(v,h)}{|V_c|} \tag{4.4}$$

Note that the fairness visibility for a given group c can be viewed as the average fairness perception of all the nodes that belong to the protected group c. For example, the group c may refer to all the papers written by well-established authors in the peer review network. To determine whether the decision function h is fair, we compare the fairness visibility for different groups of nodes using the definition below.

Definition 9 (Fairness Visibility Parity)**.** *The decision function h satisfies fairness visibility parity for V_c and $V_{c'}$ if*

$$FV(V_c) = FV(V_{c'}) \tag{4.5}$$

For example, in a peer review network, we may categorize the papers into two groups, those written by famous authors or those written by less established researchers. If the average fairness perception for both groups of papers are the same, then their fairness visibility parity holds. The larger the disparity, the more biased are the decisions as perceived by the groups.

A standard approach for measuring group fairness is to compute demographic parity, which is defined as follows:

Definition 10 (Demographic Parity)**.** *The decision function h satisfies demographic parity for V_c and V_c' if*

$$P(h(v) = 1 \mid v \in V_c) = P(h(v) = 1 \mid v \in V_c') \tag{4.6}$$

Unlike fairness visibility parity, demographic parity is computed for non-relational data since it ignores the neighborhood structure of a node. In the context of peer review network, each probability term in Eqn. (4.6) corresponds to the acceptance rate of papers that belong to the group c or c'. For brevity, we termed $P(h(v) = 1|v \in V_c)$ as the acceptance probability for the group V_c. The theorem below illustrates the relationship between fairness visibility and acceptance probability.

Theorem 2. *Assuming the network graph is connected and the decision function h has non-zero true positive and false positive rates, the fairness visibility of group V_c, based on the neighborhood peer expectation, converges to the acceptance probability for V_c as the δ-neighborhood size increases.*

Proof: Given a node v, note that $N(v) \to V$ as the δ-neighborhood expands since the network graph is assumed to be connected. Furthermore, if the true positive and false positive rates for h are non-zeros, then eventually $\mathbb{E}[h(v)] > 0, \forall v \in V$ by expansion of δ-neighborhood. It follows that $f(v, h) = 1$ if $h(v) = 1$ and $f(v, h) = 0$ if $h(v) = 0$. Thus $FV(V_c)$ converges to $P(h(v) = 1|v \in V_c)$. □

Corollary 2.1. *Given a connected network G, the decision function h satisfies demographic parity if and only if there exists a positive integer k such that for all $\delta \geq k$, fairness visibility parity holds for h with the given δ-neighborhood.*

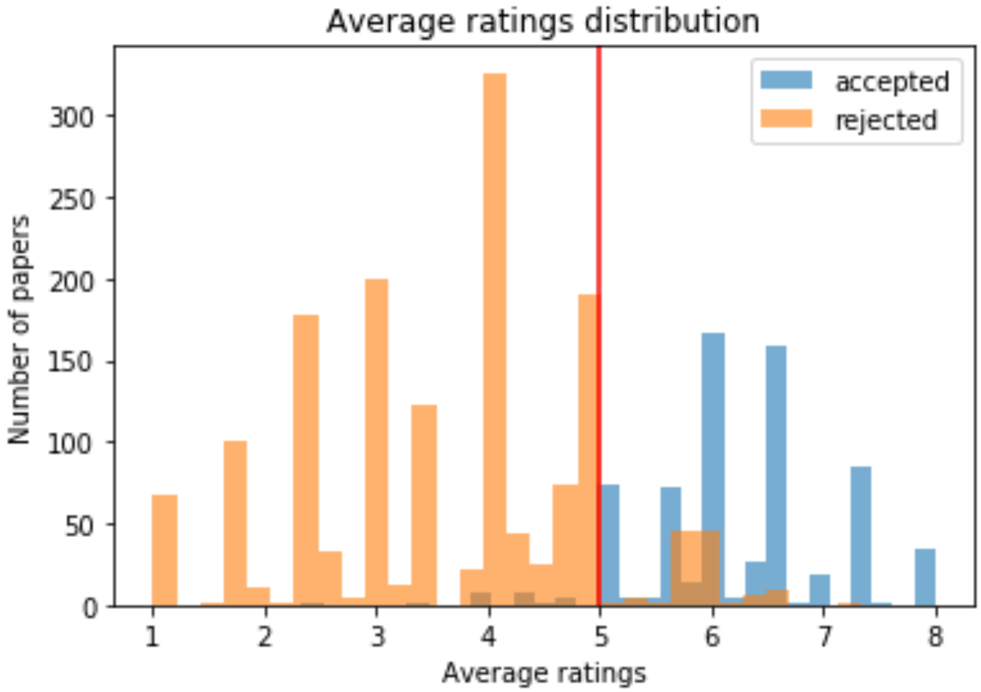


Figure 4.2: The average rating distribution of submitted papers. The red line indicates the threshold used for classifying papers as acceptable ($y = 1$) or unacceptable ($y = 0$).

4.4 Application to Peer Review Networks

This section presents a case study on the application of our proposed approach to a peer review network dataset.

4.4.1 Data

We constructed a network from the peer review dataset collected for the ICLR 2020 conference from the `OpenReview.net` website. Specifically, for each submitted paper, we gathered information about its title, abstract, list of authors and their affiliations. In addition, the anonymized reviews and acceptance decision for each reviewed paper are also available. For the ICLR 2020 conference, the number of submitted papers is 2594. However, 382 of the submissions were withdrawn. Our analysis is therefore restricted to only 2212 papers which had been reviewed. We use this information to create a network that contains 2212 nodes, one for each peer-reviewed paper.

The total number of accepted papers, either as oral or poster presentation, is 687 while the

Table 4.1: Summary distribution of acceptable and accepted papers for the ICLR 2020 conference.

		Acceptability	
		$y = 1$	$y = 0$
Acceptance	$h = 1$	589	98
Decision	$h = 0$	117	1408

(a) All papers

		Acceptability	
		$y = 1$	$y = 0$
Acceptance	$h = 1$	94	13
Decision	$h = 0$	12	153

(b) Famous authors

	Acceptability	
	$y = 1$	$y = 0$
$h = 1$	495	85
$h = 0$	105	1255

(c) Non-famous authors

		Acceptability	
		$y = 1$	$y = 0$
Acceptance	$h = 1$	190	34
Decision	$h = 0$	21	328

(d) Top institutions

	Acceptability	
	$y = 1$	$y = 0$
$h = 1$	399	64
$h = 0$	96	1080

(e) Non-top institutions

number of rejected papers is 1525. Thus, the conference acceptance rate is around 31%. We use the acceptance decision of each paper as the decision function h to evaluate fairness perception. We consider the acceptability of a paper, in terms of its average review ratings, as the target outcome y. Our assumption here is that the reviewers are rational-minded individuals, whose average ratings given to a paper reflect the technical merits and acceptability level of the paper. Figure 4.2 shows histograms of average review ratings for the accepted and rejected papers. Given that the number of accepted papers is 687, we choose an acceptability threshold of 6 since it gives a number of acceptable papers that has the closest match to the actual number of accepted papers. With this threshold, all papers whose average ratings are larger than 5 are considered acceptable, i.e., $y = 1$. Table 4.1(a) shows a confusion matrix comparing the acceptability of the paper (y) and its acceptance decision (h).

The total number of authors who had submitted papers to the conference was 6953. We were able to extract authorship information for each paper, such as names and email addresses of the co-authors, affiliation, gender, and scholarid by prepossessing the the users profile page on the OpenReview website. Based on this information we classified the submitted papers into groups based on the following “protected" attributes:

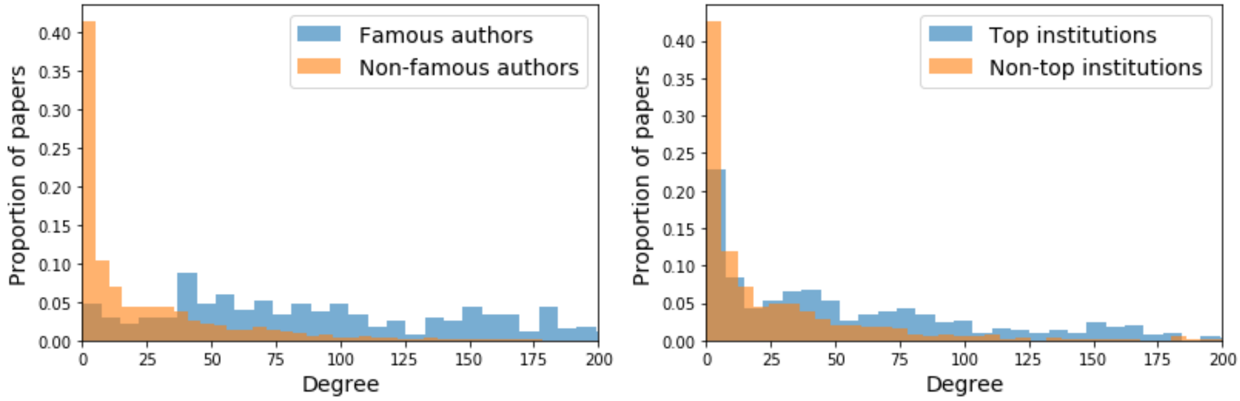
- **Famous author papers**: If a paper includes one or more famous authors, its protected attribute value is $X_p = 0$, otherwise $X_p = 1$. We consider the top 500 authors with highest h-index according to Google scholar[1] as famous authors. With this designation, 272 of the submissions were classified as famous author papers.
- **Top institution papers**: If a paper has an author from a top-10 university according to the csrankings.org website[2], then it its protected attribute value is $X_p = 0$, otherwise $X_p = 1$. We found 573 of the submitted papers have at least one author from a top institution.

A breakdown on the number of acceptable and accepted or rejected papers for each group is shown in Tables 4.1(b)-(e). The results given in these tables are consistent with previous research, which had suggested that conference paper acceptance decisions are generally biased in favor of famous authors or papers written by authors from top institutions [157, 162]. In particular, the results suggest that the chance for an acceptable paper to be accepted is significantly higher for papers written by famous authors (88.7%) or authors from a top institution (90.1%) compared to those written by non-famous authors (82.5%) or authors from lower ranked institutions (80.6%). Papers by famous or top institution authors also have a higher chance of getting their unacceptable papers accepted compared to those by non-famous authors or authors from lower ranked institutions, as reflected by their higher false positive rates.

We use the co-authorship information extracted from the authors' profile pages on OpenReview.net to construct the links between the nodes in the network. We consider two papers are linked if they share a common co-author or if the authors have collaborated in the past. Figures 4.3-(a) and 4.3-(b) show the degree distribution of the networks based on the famous author and

[1] `https://scholar.google.com/citations?view_op=search_authors&hl=en&mauthors=label:machine_learning`

[2] `http://csrankings.org/#/index?all`



(a) Famous author

(b) Top institution

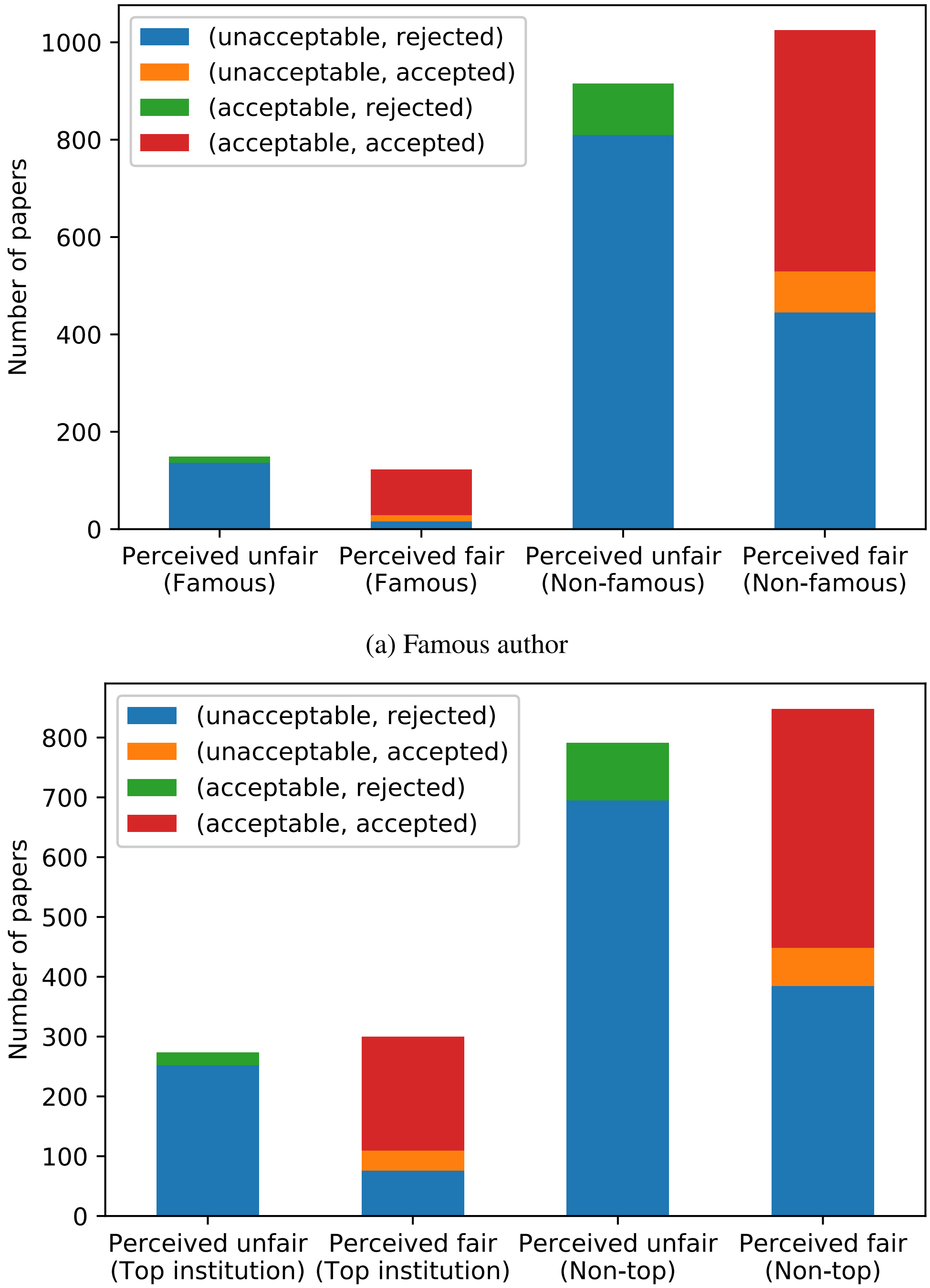
Figure 4.3: Degree distribution of nodes in peer-review network.

top institution protected attributes. The results suggest that papers by famous authors or authors from top institutions tend to have higher degree (on average) and a heavier tail in their distribution compared to those written by non-famous authors or authors from lower ranked institutions.

4.4.2 Fairness Perception

We applied the proposed fairness perception function to the network and evaluated the proportion of papers who perceived the paper acceptance decision to be fair or unfair. The results are shown in Figure 4.4-(a) for the famous author protected attribute. Despite the fact that papers by famous authors are generally favored (i.e., have higher true positive and false positive rates), the bar chart shown in Figure 4.4-(a) suggests that the majority of them still perceived the decision to be unfair. According to the fairness perception function, the main source of their discontent is the unacceptable papers that were rejected (i.e., the blue bar), which they believe should have been accepted. For papers by non-famous authors, Figure 4.4-(a) tells an opposite story as the majority of them perceived the paper acceptance decisions to be fair. Although a significantly large number of them still perceived the decision to reject their unacceptable papers as unfair (see the blue bar for perceived unfair), the non-famous author papers are more amenable to accepting the decision to reject their unacceptable papers (see the proportion of blue bar for perceived fair).

The preceding results show a potential pitfall of using the fairness perception function (with



(a) Famous author

(b) Top institution

Figure 4.4: Assessment of network-centric fairness perception

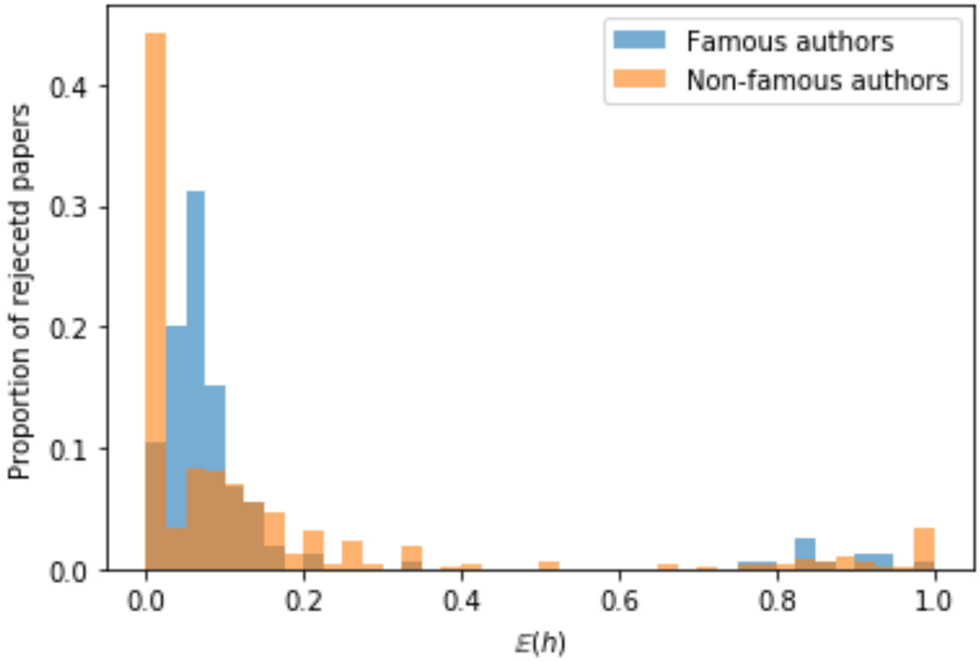


Figure 4.5: Comparison of $\mathbb{E}[h(v)]$ for rejected papers by famous and non-famous authors.

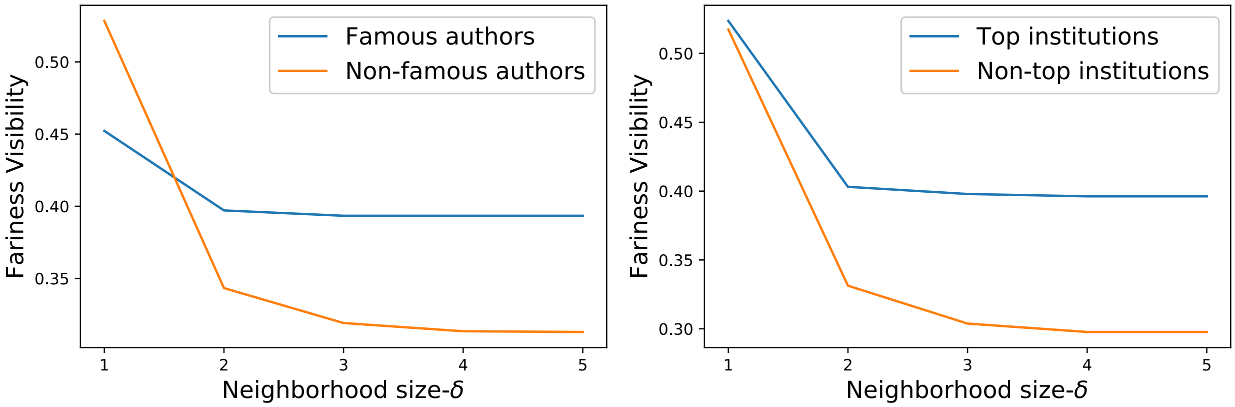
neighborhood size $\delta = 1$). Although the analysis of the confusion matrices given in Table 4.1 suggests that the decision is biased in favor of papers written by famous authors or top institutions, the non-famous authors or those from non-top-tier institutions still perceived the decisions to be fair! This can be explained as follows. Since our fairness perception function depends on the computation of $\mathbb{E}[h(v)]$, we examine the distribution of $\mathbb{E}[h(v)]$ for rejected papers by famous and non-famous authors. The results are shown in Figure 4.5. More than 40% of the rejected papers by non-famous authors have an expected value close to 0 compared to around 10% of the rejected papers by famous authors, Based on the definition given in Eqn. (4.2), the larger the proportion of papers with $\mathbb{E}[h(v)]$ close to zero, the more likely they perceived the decision to be fair. One possible explanation for the famous authors to have fewer proportion of papers with $\mathbb{E}[h(v)]$ close to zero is due to the degree distribution of their nodes (see Figure 4.3). Since papers by famous authors generally have a higher degree, this increases the number of nodes in their neighborhood, which in turn, results in a higher expected value according to the formula used to compute the neighborhood peer expectation. In contrast, many papers by non-famous authors have low degree nodes, thus producing more nodes with low $\mathbb{E}[h(v)]$.

4.4.3 Fairness Visibility

In this section, we will empirically evaluate the theoretical results for fairness visibility, which provides a possible solution to alleviate the potential pitfall of using our fairness perception function. For this experiment, we vary the δ-neighborhood size from 1 to 5 and compute the corresponding fairness visibility measure with respect to the protected attributes. The results are plotted in Figure 4.6.

For Figure 4.6-(a), observe that the fairness visibility of papers by famous authors are initially lower than that for papers by non-famous authors when $\delta = 1$. This means that, on average, the papers by famous authors have lower perceived fairness. As δ increases, fairness visibility decreases for both groups of papers. However, the rate of decrease is higher for papers by non-famous authors. According to Theorem 2, under mild assumption, fairness visibility will converge to the acceptance probability of each subgroup of the protected attribute when δ increases. Since the acceptance probability for $X_p = 0$ (famous authors) is higher than that for $X_p = 1$ (non-famous authors), the fairness visibility for famous authors will be higher for larger values of the neighborhood size, δ. This provides a strategy to counter against the potential pitfall of using fairness perception by expanding the neighborhood size δ. Furthermore, it is worth noting that the peer review network is not a connected graph. As a result, the fairness visibility does not converge exactly to the acceptance probability for each group, which is 0.2989 (for non-famous authors) and 0.3933 (for famous authors), when δ is sufficiently large.

A similar observation can be made when analyzing the effect of increasing neighborhood size on fairness visibility using top institution as protected attribute. As shown in Figure 4.6-(b), increasing δ leads to lower fairness visibility. However, with sufficiently large δ, the fairness visibility for papers by authors from top institutions is higher than that for papers by authors from lower ranked institutions. By setting $\delta = 2$, the fairness visibility provides a good assessment on the true bias of the paper acceptance decisions.



(a) Famous author

(b) Top institution

Figure 4.6: Effect of neighborhood size on fairness visibility.

4.5 Conclusion

This paper presents a novel approach for algorithmic fairness in network data. Motivated by the equity theory in social science, we introduced the concept of fairness perception as a local formulation of fairness and quantified this notion through an axiomatic approach to analyze its properties. We also showed how our proposed network-centric fairness perception function can be extended to a group fairness measure known as fairness visibility. We provided theoretical analysis to demonstrate its relationship to demographic parity. Using a peer-review network as case study, we also examined its utility in terms of assessing the perception of fairness in paper acceptance decisions. We also highlighted a potential pitfall of using fairness visibility measure as it can be exploited to mislead individuals into perceiving that the algorithmic decisions are fair. Finally, we show how to alleviate the problem by increasing the local neighborhood size.

CHAPTER 5

MITIGATING PERCEPTION OF BIAS IN PEER-REVIEW DECISION MAKING

This chapter presents a novel recommendation algorithm that considers both the ratings provided by the reviewers and the co-authorship network. Specifically, the proposed algorithm balances the trade-off between maximizing review ratings and minimizing the perception of bias in the peer review decisions. Motivated by the work on fairness perception in previous chapter, we provide theoretical bounds on the gap between the average rating for different groups of individuals when fairness perception is maximized. Finally, we demonstrate the effectiveness of the proposed algorithm using open peer review data for a premier conference in machine learning.

5.1 Introduction

Peer review has an undeniable role in the academic advancement of scholars. It helps to assess whether the quality of a submitted work merits acceptance for conference or journal publication. It is also commonly used to evaluate grant proposals submitted for funding allocation. Despite its benefits, similar to other subjective approaches, the peer review system suffers from potential biases among the reviewers [87]. For example, previous studies have suggested that conference peer review systems tend to be biased in favor of papers written by prominent authors or authors from well-known institutions [162]. In order to mitigate this problem, researchers have begun investigating algorithmic-driven approaches to improve different stages of the peer review process. This includes learning to order the submissions for display during the reviewers' bidding step [56], matching the submission to appropriate reviewers during the reviewer assignment step [76], and calibrating the scores provided by reviewers who have varying degrees of leniency [170].

Despite these efforts, there are still concerns regarding fairness of the peer review decisions, especially in top-tier computer science conferences, as the number of submissions has grown significantly in recent years. The goal of this paper is to develop a recommendation algorithm that identifies highly-rated papers for acceptance by learning a ranking function that takes into account acceptability of the papers and fairness of the decision outcomes. Here, the notion of fairness considers both disparity in the decision outcomes of similarly qualified papers by different groups of authors as well as the authors' perception of fairness in the decisions.

The perception of fairness is important in peer review process as it affects credibility of the conference or journal. In particular, if the decisions were viewed as favoring certain groups of privileged individuals at the expense of others, this may discourage other authors from submitting their work to the same venue again in the future. Despite its importance, quantifying and optimizing the perceived fairness remains a challenge as most conference management systems do not provide an option for authors to give their feedback regarding the fairness of the paper acceptance decisions. While some conferences provide a rebuttal phase to solicit author's feedback, this was often used as a mechanism to clarify or respond to comments by reviewers.

This chapter is built on development of a fairness perception criteria in chapter four. However, the fairness perception measure alone is insufficient for two reasons. First, maximizing the measure does not provide guarantees of global fainess for all groups of individuals since the criteria is defined based on local neighborhood information only. This limitation was indeed noted in chapter four, we showed that their measure may mislead individuals into perceiving the overall decisions to be fair even though the decisions are still biased towards certain groups of individuals. Second, the proposed fairness perception measure was designed for classification instead of recommendation problems. It requires access to ground truth labels to train a model to predict the decision outcome. For recommendation problems, this assumption is no longer valid, and thus, requires a new formulation to define a continuous-valued score for ranking the papers.

In this chapter, we present a fairness-aware recommendation algorithm for mitigating the perception of bias in peer reviews. Specifically, given an initial set of review ratings and a co-authorship network, our goal is to provide a ranking of the papers based on a combination of factors, including fairness perception and statistical parity, which is a measure of group-level fairness. We examine how to improve recommendation of papers for acceptance by balancing the trade-off between maximizing fairness perception and minimizing statistical disparity between different groups of individuals. Using a graph-theoretic approach, we provide theoretical bounds on the gap in statistical parity of the decisions when fairness perception is maximized for all the groups and show that the bounds are potentially large, depending on the network structure and how the nodes belonging to the protected group are distributed in the network. A novel multi-objective optimization algorithm is therefore proposed to overcome this problem. Finally, we conducted extensive experiments on real-world data from a multi-year computer science conference to demonstrate the effectiveness of our proposed algorithm in terms of the accuracy, fairness perception, and statistical parity of its recommendations.

5.2 Related Work

Algorithmic fairness is an important topic that has attracted considerable interest in recent years [16, 49, 69, 97]. Fairness can be defined at individual level as the absence of any prejudice or favoritism towards an individual with certain characteristics in a decision-making task [124]. In other words, similar people should be treated similarly [49]. For group fairness, the individuals are divided into disjoint groups on the basis of their protected attribute values [16]. Given the decisions made on individuals belonging to each group, a confusion matrix can be created from which fairness metrics are computed to statistically compare the groups. If the decisions are fair, then the metrics for different groups are expected to be equal or close to each other [53, 69, 49, 53]. Examples of group fairness metrics include *demographic parity* or *statistical parity* [49], *equalized odds* [69], and *calibration fairness* [36], which have been used in previous studies [113, 80, 78, 50, 24] to ensure the independence between the decision outcome (or risk score) and group membership of the protected attribute. However, recent studies have shown that existing metrics are not always compatible with each other [33, 58]. Kearns et al. [85] described another challenge known as *gerrymandering*, in which a classifier may appear to be fair with respect to each protected attribute separately, but not when defined jointly over multiple protected attributes. There have been several recent attempts to address these limitations. For example, Hebert-Johnson et al. [70] introduced a multi-calibration approach, which aims to guarantee the statistical fairness definitions would hold for an exponential or infinite class of groups defined by some class of functions with bounded complexity.

In addition to defining fairness metrics, methods for de-biasing decision outcomes from machine learning models have also been developed. As the input data itself is potentially biased, preprocessing methods have been developed to alleviate this problem [186, 113, 116]. For example, Louizos et al. [113] employed variational auto-encoders to learn a latent representation of the data that is independent of sensitive attributes in the data. Similarly, Zemel et al. [186] presented an algorithm for learning fair representation that preserves as much information as possible about the original data except for the protected group membership information. However, these approaches are mostly designed for *independent and identically distributed* (i.i.d.) data. To alleviate this problem, in chapter three we introduced a dyadic-level fairness criterion based on a modified network modularity measure and showed how it can be utilized for network link prediction tasks to overcome the filter bubble problem.

The concept of fairness has also been extensively investigated in other fields such as social science and criminal justice [143, 153]. These include studies that focus on understanding how individuals perceive the notion of fairness. For example, Saxena et al. [151] investigated public opinions on 3 fairness measures developed by computer science researchers and concluded that there is a preference for calibration fairness. Lee and Baykal [102] also investigated users' perception on algorithmic fairness, though their results suggest that one-third of the participants felt the algorithmic decisions to be unfair due to their failure to account for multiple concepts of fairness.

Ensuring fairness in peer review of scholarly research is important as it can have an adverse affect on the quality of the decisions as well as reputation of the organization conducting the review process (e.g., journals, conferences, or funding agencies). Recent years have witnessed growing attempts to improve the different stages of the peer review process. For example, Charlin and Zemel [26] proposed an approach for automated reviewer assignment of conference papers known as the Toronto paper matching system. Fiez et al.[56] argued that the ordering of conference papers presented to reviewers for bidding has a significant impact on the number of bids received for each paper, which in turn, affects their quality of reviews. Other similar works focusing on improving the reviewer assignment process include [158, 93]. A long-standing debate regarding peer review process is whether to reveal the identities of the authors (single blind review) and reviewers (double blind review). Tomkins et al. [162] conducted an experimental study by partitioning entire pool of reviewers for a conference into single blind and double blind. They concluded that single blind system tends to be biased in favor of papers authored by famous authors or those from top institutions. However, a subsequent study by Stelmakh et al. [156] showed the limitations of the test procedures used in [162] and designed an approach to overcome these limitations. Finally, calibration of the reviewers scores is another research direction that has been pursued recently [170] to account for the implicit biases in the ratings as some reviewers are more lenient or stringent than others.

5.3 Preliminaries

This section formalizes our problem statement and reviews some of the fairness measures available.

5.3.1 Problem Statement

Let $G =< V, E, X >$ be an attributed network, where V is the set of nodes, $E \subseteq V \times V$ is the set of links, and X is the set of node attributes. For paper acceptance recommendation, we assume each node is a paper, each link represents a co-authorship relation between two papers, and the node attributes $X = (X^{(p)}, X^{(u)})$ is a combination of the protected attributes, $X^{(p)}$, and other attributes, $X^{(u)}$. The protected attribute indicates whether the paper was written by well-known researchers (**famous authors**) or authors from highly-ranked universities or research labs (**top institutions**). If so, then $X^{(p)} = 0$, otherwise $X^{(p)} = 1$. Our rationale for using these criteria as protected attribute is based on the results from chapter four and [162], suggesting that peer review decisions tend to favor famous authors or those from well-known institutions. The attributes in $X^{(u)}$ include the review ratings and other assessment criteria associated with each paper. We assume there exists an evaluation function $\phi : X_v^{(u)} \rightarrow \mathbb{R}$ such that $s_v = \phi(X_v^{(u)})$ represents acceptability of the paper v. For example, s_v may correspond to the average reviewer ratings (or a weighted average ratings based on the reviewers' confidence) of a paper. In addition, we also consider a kernel function, $K : X^{(u)} \times X^{(u)} \rightarrow \mathbb{R}_+$, which measures the similarity between a given pair of papers. In this chapter, we use the following Gaussian radial basis function as our kernel function: $K(u, v) = \exp\left(-\frac{\|u-v\|^2}{2\sigma^2}\right)$, where σ is a hyperparameter.

The co-authorship relation is established by examining whether two papers share the same authors or have authors who had previously collaborated together in the past. Their links are represented by an adjacency matrix, A, where $A_{ij} = 1$ if papers i and j has a co-authorship relation and 0 otherwise.

Definition 11 (Peer-review recommendation)**.** *Given a co-authorship network $G =< V, E, X >$, the peer-review recommendation task is to learn a ranking function $h : V \rightarrow [0, 1]$ based on the link structure E and node attributes $X = (X^{(p)}, X^{(u)})$.*

The output of the ranking function can assist decision makers (i.e., conference program chairs) by providing an initial list of top-k highest-ranked papers recommended for acceptance. Our goal is to provide accurate recommendation that minimizes implicit biases in the peer review decisions while maximizing fairness perception of the authors. While the discussion presented in this chapter focuses on peer review recommendation, in principle, the proposed approach is applicable to other network-based recommendation systems [114, 30].

5.3.2 Fairness Measures

Quantifying the notion of fairness is one of the pillars of ethical machine learning. A widely used criterion for assessing fairness is *demographic parity* or *statistical parity* (SP) [49]. Demographic parity constrains the output of the classification problem to be independent of the protected attribute. Particularly a predictor satisfies demographic parity if there is no correlation between the protected feature and the output. In other words, demographic parity seeks to equalize the positive decision across different groups of the protected feature. Assuming both the target outcome $\hat{Y}$ and protected attributes are binary-valued, demographic parity can be formalized as follow:

$$P(\hat{Y} = 1|X^{(p)} = 0) = P(\hat{Y} = 1|X^{(p)} = 1) \tag{5.1}$$

Another well-known fairness criterion is *equalized odds* [69], which seeks to ensure that the predictions are conditionally independent of the protected attribute given the true outcome. Suppose $Y = 1$ is the advantaged outcome, such as job offer or college admission. A classifier, $\hat{Y}$, satisfies the equality of odds with respect to the protected attributed $X^{(p)}$ and binary outcome Y if both conditions below hold:

$$P(\hat{Y} = 1|X^{(p)} = 0, Y = 1) = P(\hat{Y} = 1|X^{(p)} = 1, Y = 1)$$

$$P(\hat{Y} = 1|X^{(p)} = 0, Y = 0) = P(\hat{Y} = 1|X^{(p)} = 1, Y = 0)$$

The former equation, also known as *equal opportunity* [69], refers to equality of true positive rates for the two groups whereas the latter refers to equality of false positive rates.

5.4 Methodology

This section presents our approach for mitigating bias in peer review recommendation. We first introduce the proposed neighborhood expectation function for recommendation problems and then present a simple algorithm that maximizes fairness perception. We examine its impact on the acceptance rates for different groups and theoretically show that the gap in acceptance rates can be potentially large, which motivates us to develop a multi-objective optimization algorithm that balances the two criteria.

5.4.1 Proposed Expected Function

Instead of using Equation (4.3), which requires availability of the ground truth label y, we consider the following approach to compute the neighborhood-based expectation $\mathbb{E}[h(v)]$ for our recommendation algorithm:

$$\mathbb{E}[h(v)] = \frac{\sum_{u \in N(v)} h(u)K(v, u)}{\sum_{u \in N(v)} K(v, u)} \tag{5.2}$$

Intuitively, the expected value of h is computed based on the weighted average value of neighboring nodes in the network, taking into account the node similarity values. Unlike chapter four, we consider h as a continuous-valued ranking function instead of a binary decision function.

Lemma 3. *The proposed expected value in Equation* (5.2) *is a well-behaved function, satisfying properties P1-P3.*

Proof. For the first property, since $\forall u \in N(v) : h(u) = h'(u)$, therefore $\mathbb{E}[h(v)] = \mathbb{E}[h'(v)]$ since K is independent of the ranking function h in Equation (5.2). The second property holds given the fact that the co-domain of the kernel function K is non-negative. Assuming $\forall u \in N(v) : h(u) \leq h'(u)$, we have

$$\sum_{u \in N(v)} h(u)K(v,u) \leq \sum_{u \in N(v)} h'(u)K(v,u)$$

The proof for property P2 follows since the denominator of the expected value in Equation (5.2) is the same for h and h'. The third property also holds based on the premise that the induced subgraphs G_u and G_v are isomorphic with respect to h. This guarantees that the h and K values for all the nodes the neighborhood of u and v are the same. As a consequence, $\mathbb{E}[h(u)] = \mathbb{E}[h(v)]$, which completes the proof. □

5.4.2 Maximizing Fairness Perception

Let $\mathbb{C} = \{C_1, \ldots, C_k\}$ be the set of all connected components in G. The theorem below presents a simple solution for the ranking function h that maximizes fairness perception.

Theorem 4. *Given a ranking function h and a network $G =< V, E, X >$, the function achieves its maximum fairness perception, i.e., $\forall v \in V : f(v,h) = 1$, if and only if $\forall C \in \mathbb{C} : h(v) = h(v')$ for all $v, v' \in C$.*

Proof. If $h(v)$ is identical for all the nodes v in the same connected component, then according to Definition 5.2, it is easy to see that $\forall v : \mathbb{E}[h(v)] = h(v)$. As a result, the fairness perception function $f(v,h) = 1$ for all $v \in C$. Since this property holds for all $C \in \mathbb{C}$, therefore $\forall v \in V : f(v,h) = 1$.

Next, we will show that if $\forall v \in V : \ f(v,h) = 1$, then all the nodes v in the same connected component must have the same value for $h(v)$. This is trivially satisfied if the connected component has only one node. Thus, we consider the case for connected components with at least two nodes. By contradiction, assume that fairness perception is maximized and there exists a connected component $C \in \mathbb{C}$ containing nodes with different rank values h. Let $C_{min} = \{u \in C | h(u) = m\}$ where $m = min\{h(u)|u \in C\}$. Since the rank values are not uniform, there must exist a node $u_m \in C_{min}$ connected to another node $v \in C \setminus C_{min}$, in which $h(u_m) \neq h(v)$. If there is no such a u_m, then C_{min} must not be connected to $C \setminus C_{min}$, which cannot be true. Thus, $\forall v \in N(u_m) : h(u_m) \leq h(v)$ and there exist a neighboring node $v \in N(u_m)$ such that $h(v) > h(u_m)$. Using the expected value function defined in Equation (5.2), we have

$$\begin{aligned}\mathbb{E}[h(u_m)] &= \frac{\sum_{v \in N(u_m)} h(v) K(u_m, m)}{\sum_{v \in N(u_m)} K(u_m, v)} \\ &> \frac{\sum_{v \in N(u_m)} h(u_m) K(u_m, v)}{\sum_{v \in N(u_m)} K(u_m, v)} \\ &= h(u_m)\end{aligned}$$

Since $h(u_m) < \mathbb{E}[h(u_m)]$, $f(u_m, h) = 0$, which contradicts the assertion that fairness perception is maximized. Thus, the original assumption that the connected component has nodes with different rank values must be wrong. □

Theorem 4 suggests a trivial solution for maximizing fairness perception is by assigning the same rank value h to every node in a connected component. However, maximizing fairness perception alone is insufficient for several reasons. First, it does not guarantee good performance as all the nodes (papers) in the same connected component have identical rank values irrespective of their acceptability score s. Second, since fairness perception is an individual-level metric based on local neighborhood features rather than global features, it does not guarantee similarly qualified papers by different groups of authors will be treated equally.

One way to measure equity across the different groups is to apply a group-level fairness criterion such as statistical parity (see Equation (5.1)), which was originally introduced for classification problems. Here, we propose the following weighted statistical disparity (WSD) metric to determine the disparity in the average rank scores h of similarly qualified papers for the different groups of authors. Specifically, we first partition the acceptability scores of the papers into a set of K ordinal values, $\hat{s}_1 < \hat{s}_2 < \cdots < \hat{s}_K$. For example, if the average rating of a paper v ranges between 0 to 10, then $\hat{s}_i$ may correspond to the rounded integer value of s_v, $r(s_v) = \hat{s}_i$. Let $B_i^{(j)} = \{v \in V \mid r(s_v) = \hat{s}_i, X_v^{(p)} = j\}$ denotes the set of papers with similar acceptability scores, $\hat{s}_i$, and $\bar{h}_i^{(j)} = \frac{1}{|B_i^{(j)}|} \sum_{v \in B_i^{(j)}} h(v)$ denotes the average rank h of such papers from protected group $X^{(p)} = j$. The following equation is used to measure the weighted statistical disparity across the different groups:

$$\Gamma(h) = \frac{1}{K} \sum_{i=1}^{K} \hat{s}_i \big(\bar{h}_i^{(0)} - \bar{h}_i^{(1)}\big)^2 \tag{5.3}$$

Note that $\Gamma(h) = 0$ if the average rank of similarly qualified papers are identical for the different groups of the protected attribute. The measure also emphasizes on ensuring fairness in the ranking of papers with higher acceptability scores since such papers are the ones more likely to be recommended for acceptance. The theorem below illustrates the relationship between fairness perception and the weighted statistical disparity measure.

Theorem 5. *Given a ranking function h that maximizes fairness perception, i.e.,* $\forall v \in V : f(v, h) = 1$*, its weighted statistical disparity satisfies the following inequality:* $\kappa \times max\Big\{\delta^*, 0\Big\} \le \Gamma(h) \le \kappa \times \Delta^*$*, where* $\kappa := \frac{1}{K} \sum_{i=1}^{K} \hat{s}_i$*,*

$$\begin{aligned} \delta^* &= sign\Big((h_{\min}^{(0)} - h_{\max}^{(1)})(h_{\max}^{(0)} - h_{\min}^{(1)})\Big) \\ &\quad \times \min\Big\{(h_{\min}^{(0)} - h_{\max}^{(1)})^2, (h_{\max}^{(0)} - h_{\min}^{(1)})^2\Big\}, \\ \Delta^* &= \max\Big\{(h_{\min}^{(0)} - h_{\max}^{(1)})^2, (h_{\max}^{(0)} - h_{\min}^{(1)})^2\Big\}. \end{aligned}$$

Here, we have denoted $h_{\min}^{(j)} = \min\{h(v) \mid X_v^{(p)} = j\}$ *and* $h_{\max}^{(j)} = \max\{h(v) \mid X_v^{(p)} = j\}$.

Proof. First, note that $h_{\min}^{(j)} \le \bar{h}_i^{(j)} \le h_{\max}^{(j)}$ since

$$\bar{h}_i^{(j)} = \frac{1}{|B_i^{(j)}|} \sum_{v \in B_i^{(j)}} h(v) \le \frac{1}{|B_i^{(j)}|} \sum_{v \in B_i^{(j)}} h_{\max}^{(j)} = h_{\max}^{(j)}$$

$$\bar{h}_i^{(j)} = \frac{1}{|B_i^{(j)}|} \sum_{v \in B_i^{(j)}} h(v) \geq \frac{1}{|B_i^{(j)}|} \sum_{v \in B_i^{(j)}} h_{\min}^{(j)} = h_{\min}^{(j)}$$

Using these inequalities, it can be easily shown that

$$h_{\min}^{(0)} - h_{\max}^{(1)} \leq \bar{h}_i^{(0)} - \bar{h}_i^{(1)} \leq h_{\max}^{(0)} - h_{\min}^{(1)} \text{ and}$$

$$(\bar{h}_i^{(0)} - \bar{h}_i^{(1)})^2 \leq \max\left\{(h_{\min}^{(0)} - h_{\max}^{(1)})^2, (h_{\max}^{(0)} - h_{\min}^{(1)})^2\right\}$$

Replacing the above inequality into Equation (5.3) and using the definition of Δ^*, we obtain the following upper bound:

$$\begin{aligned}\Gamma(h) &\leq \frac{1}{K} \sum_i \hat{s}_i \max\left\{(h_{min}^{(0)} - h_{max}^{(1)})^2, (h_{max}^{(0)} - h_{min}^{(1)})^2\right\} \\ &= \kappa \times \Delta^*\end{aligned}$$

To obtain the lower bound for $\Gamma(h)$, we use the following:

$$\begin{aligned}(\bar{h}_i^{(0)} - \bar{h}_i^{(1)})^2 \geq \text{sign}\Big((h_{\min}^{(0)} - h_{\max}^{(1)})(h_{\max}^{(0)} - h_{\min}^{(1)})\Big) \\ \times \min\left\{(h_{\min}^{(0)} - h_{\max}^{(1)})^2, (h_{\max}^{(0)} - h_{\min}^{(1)})^2\right\} = \delta^*\end{aligned}$$

If $(h_{\min}^{(0)} - h_{\max}^{(1)})$ and $(h_{\max}^{(0)} - h_{\min}^{(1)})$ have opposite signs, then δ^* is negative. Since $(\bar{h}_i^{(0)} - \bar{h}_i^{(1)})^2$ is non-negative, we can obtain a tighter bound as follows:

$$(\bar{h_i}^{(0)} - \bar{h_i}^{(1)})^2 \geq \max\left\{\delta^*, 0\right\} \tag{5.4}$$

Replacing the above inequality into Equation (5.3) yields

$$\Gamma(h) \geq \frac{1}{K} \sum_i \hat{s}_i \max\left\{\delta^*, 0\right\} = \kappa \times \max\left\{\delta^*, 0\right\}$$

which completes the proof. □

The preceding theorem shows that maximizing fairness perception does not guarantee parity in the recommended decisions for different groups. The latter also depends on the distribution of the protected attribute values across the network, which is beyond the control of the decision-makers. Thus, it would be desirable to find a compromise that considers the trade-off between maximizing fairness perception and minimizing the weighted statistical disparity. In the next section, we describe our algorithm to achieve this balance.

5.4.3 FPRank: Fair Peer Review Recommendation

This section presents **FPRank**, our proposed recommendation algorithm, which learns a ranking function h that considers the acceptability scores of the papers and the trade-off between maximizing fairness perception and minimizing the weighted statistical disparity measure, Γ.

First, we re-write the expected value given in Equation (5.2) as $\mathbb{E}[h(v_i)] = \sum_j W_{ij} h(v_j)$, where $W = D^{-1}(K \odot A)$ and $\odot$ denote a Hadamard product between the adjacency matrix A and the kernel matrix K. D is a diagonal matrix, in which $D_{ii} = \sum_j [K \odot A]_{ij}$. A trivial approach to learn a ranking function that maximizes fairness perception is to solve $h = Wh$, or equivalently, $(W - I)h = 0$. The solution is obtained by finding the null space of the matrix $W - I$, which admits the null vector $h = \mathbf{0}$ as its trivial solution. In addition to maximizing fairness perception, the ranking function h should also match acceptability of the paper as given by s. In other words, the rank of the paper should be high if its acceptability score is high. Thus, one of our objectives is to minimize the discrepancy between s and h. Furthermore, the ranking function h must also be trained to minimize the weighted statistical disparity, $\Gamma(h)$, to ensure it is not biased against certain groups of nodes.

Putting everything together, the objective function to be solved by our framework is as follows:

$$\begin{aligned} \text{argmin}_h \quad & ||s - h||^2 + \alpha\Gamma(h)^2 + \beta||z||^2 \\ \text{s.t.} \quad & \forall i : \sum_j W_{ij} h_j \leq h_i + z_i, \;\; h_i \geq 0, z_i \geq 0 \end{aligned} \tag{5.5}$$

z is a slack variable that relaxes the constraint for fairness perception to be equal to 1 for all the nodes. The hyperparameters α and β control the tradeoff between matching h to the paper acceptability scores, ensuring group parity, and minimizing the number of constraint violations. During training, our algorithm finds the values of α and β that minimizes the power mean of the three terms in the objective function[1]. The objective function can be solved using a standard quadratic programming package such as CVX [46, 6].

[1] The power mean for $x_1, \ldots, x_n$ is $M_p(x_1, \ldots, x_n) = \left(\frac{1}{n}\sum_{i=1}^n x_i^p\right)^{\frac{1}{p}}$ where p is a hyper parameter.

Table 5.1: Summary statistics of submitted and accepted papers for the ICLR conference from 2017 to 2020.

Year	# papers	# Accepted papers	% Top Institution	% Famous authors
2017	488	243	25%	22%
2018	402	229	33%	17%
2019	1419	502	27%	15%
2020	2212	687	26%	12%

5.5 Experimental Evaluation

This section describes the experiments performed to investigate the effectiveness of our fairness-aware recommendation algorithm.

5.5.1 Experimental Setup

5.5.1.1 Data

We consider a multi-year peer review data from a major computer science conference (ICLR) for our experiments. The dataset was scraped from the `OpenReview.net` website, which provides an API to access the list of submitted papers as well as their meta-data, including reviews, comments, and final decisions. Table 5.1 shows the summary statistics of the submitted and accepted papers from 2017 to 2020. We extracted the authorship information for each paper, including the authors names, affiliations, and scholarid. We also identified the email addresses of their co-authors by preprocessing the authors' profile pages on the OpenReview website. Similar to the approach used in previous chapter, we categorized the submitted papers into different groups based on the values of two protected attributes—*famous author papers* and *top institution papers*. We used Google Scholar to identify famous authors by examining the top-500 researchers with the highest h-index scores[2]. Thus, if a paper includes one or more famous authors, then $X^{(p)}_{\text{author}} = 0$, otherwise $X^{(p)}_{\text{author}} = 1$. To identify the top institutions, we used the rank list provided by csrankings.org[3]. If

[2] `https://scholar.google.com/citations?view_op=search_authors&hl=en&mauthors=label:machine_learning`

[3] `http://csrankings.org/#/index?all`

a paper has an author from a top-10 highest ranked university, then $X_{\text{inst}}^{(p)} = 0$, otherwise $X_{\text{inst}}^{(p)} = 1$. The co-authorship relation was established by combining the list of co-authorship information provided on the user profile page on OpenReview website with the co-authorship information gathered from the DBLP website[4]. We consider the acceptability score of a paper s in terms of its average review ratings.

5.5.1.2 Baseline Algorithms

We compare our proposed FPRank method against the following baselines: (1) Calibrated equalized odds postprocessing, **CEP** [145], which is a method for calibrating the classifier output scores to ensures fairness in terms of equalized odds measure. We use a variation of the method to ensure equality of false positive rates **CEP (FPR)**. (2) Reject option classification, **ROC** [79], which is a post-processing method that guarantees fairness in terms of statistical parity **ROC (SPD)** or equal opportunity **ROC (EOD)**. (3) **FairTop-k** [185], which is a fairness-aware ranking algorithm that identifies a subset of k candidates from a large pool of candidates by selecting the best candidates based on a ranked group fairness criterion. We use the implementation provided by AI Fairness 360 software[5] for the CEP and ROC baselines. We also consider a variation of our method, **Max Perception**, which maximizes only fairness perception and consistency with acceptability score s without considering statistical parity (i.e., see Equation 5.6 without $\Gamma(h)$ and the slack variables).

5.5.1.3 Evaluation Metrics

We consider the conference decision as the ground truth outcome and calculate the widely-used average precision measure of each algorithm based on their recommended papers, i.e., $\frac{1}{m}\sum_k P@k$, where m is the number of accepted papers and $P@k$ is the precision based on the top-k recommended papers. We also compute its overall fairness perception by taking the average fairness perception of the papers, i.e., $\frac{1}{|V|}\sum_{v\in V} f(v,h)$. The metric ranges between 0 and 1, with larger values suggest a

[4] `https://dblp.uni-trier.de/xml/`

[5] `https://aif360.mybluemix.net`

Table 5.2: Average precision of various methods compared to the conference decisions (using top institutions as protected attribute).

	2017	2018	2019	2020
FPRank	**0.9246**	**0.8333**	0.7956	**0.9093**
Max Perception	0.6901	0.6979	0.4601	0.3063
ROC (SPD)	0.7687	0.7789	0.7830	0.7381
ROC (EOD)	0.7769	0.7794	0.7797	0.7611
CEP (FPR)	0.5606	0.6934	0.5074	0.6705
FairTop-k	0.9004	0.8288	**0.8810**	0.8464

Table 5.3: Average precision of the various methods compared to the conference decisions (using famous authors as protected attribute).

	2017	2018	2019	2020
FPRank	**0.9262**	**0.8356**	0.7956	**0.9085**
Max Perception	0.6901	0.6979	0.4601	0.3063
ROC (SPD)	0.7738	0.7575	0.7684	0.7467
ROC (EOD)	0.7769	0.7794	0.7797	0.7611
CEP (FPR)	0.5641	0.6976	0.5074	0.6705
FairTop-k	0.8971	0.8263	**0.8790**	0.8487

higher perception of fairness in the decision. Finally, we evaluate the weighted statistical disparity of each method using Equation 5.3. To compute the metric, the acceptability scores s_i were discretized into $\hat{s}_i$ by rounding the average review ratings to the nearest integer value.

5.5.2 Experimental Results

The first goal of our experiments is to compare the rankings generated by the different methods against the conference decisions. Table 5.2 summarizes the average precision of FPRank and other baseline methods using top institution as protected attribute while Table 5.3 shows the corresponding results using famous author as protected attribute. The results in both tables suggest that FPRank outperforms all the baseline methods in 3 out of 4 years. Although its precision is lower than FairTop-K in 2019, it outperforms FairTop-K in terms of average precision and other fairness metrics to be described below in all other years. Observe that the Max Perception baseline has a lower average precision than FPRank, which suggests that maximizing fairness perception does not guarantee high quality recommendation.

Table 5.4: Comparison of average fairness perception (using top institutions as protected attribute)

	2017	2018	2019	2020
FPRank	0.8709	0.8682	0.7463	0.5023
Max Perception	**0.9918**	**0.9577**	**0.8118**	0.5054
ROC (SPD)	0.7766	0.8159	0.6758	0.5027
ROC (EOD)	0.8340	0.8159	0.6892	0.4973
CEP (FPR)	0.7889	0.8358	0.6723	0.4991
FairTop-k	0.8340	0.8184	0.6794	0.4986
Conference Decision	0.8135	0.8308	0.6963	**0.5077**

Table 5.5: Comparison of average fairness perception (using famous authors as protected attribute)

	2017	2018	2019	2020
FPRank	0.8709	0.8607	0.7449	0.5023
Max Perception	**0.9918**	**0.9577**	**0.8118**	0.5054
ROC (SPD)	0.8094	0.8159	0.6737	0.4977
ROC (EOD)	0.8340	0.8159	0.6892	0.4973
CEP (FPR)	0.8197	0.8308	0.6723	0.4991
FairTop-k	0.8320	0.8134	0.6758	0.4973
Conference Decision	0.8135	0.8308	0.6963	**0.5077**

Next, we compare the average fairness perception of the different methods. Table 5.4 summarizes the results for top institution as protected attribute. Except for 2020, Max Perception has the highest fairness perception, which is not surprising since the algorithm explicitly tries to maximize the measure. FPRank has the second highest fairness perception values and consistently outperforms other baseline methods including FairTop-K. Similar conclusions can be reached for the results shown in Table 5.5 for famous authors as protected attribute. For 2020, it appears that the original conference decision has the highest fairness perception, though it is quite close to other methods and is lower than other years.

Finally we compare the performance of the various methods in terms of their group fairness metric, weighted statistical disparity. Tables 5.6 and 5.7 summarize the results using top institution and famous author, respectively, as the protected attribute. Observe that FPRank consistently achieves weighted statistical disparity values close to 0 in all years, which suggests that its recommendation is not biased against one of the protected groups. These results support our claim that FPRank can achieve high average precision in its recommendation while balancing the trade-off between

Table 5.6: Comparison of weighted statistical disparity (using top institutions as protected attribute)

	2017	2018	2019	2020
FPRank	**0.0657**	**0.0049**	0.0040	**0.0000**
Max Perception	0.3707	0.1831	0.0165	0.0003
ROC (SPD)	0.3619	0.1649	0.0065	0.0101
ROC (EOD)	0.3466	0.1637	0.0009	**0.0000**
CEP (FPR)	0.6164	0.1637	**0.0000**	0.0001
FairTop-k	0.3459	0.1644	0.0040	0.0001
Conference Decision	0.3464	0.0110	0.0012	0.0023

Table 5.7: Comparison of weighted statistical disparity (using famous authors as protected attribute)

	2017	2018	2019	2020
FPRank	**0.0709**	0.0391	0.0044	**0.0000**
Max Perception	0.2835	0.1798	0.0557	0.0004
ROC (SPD)	0.2375	0.1753	0.0102	0.0103
ROC (EOD)	0.1825	0.1638	**0.0002**	**0.0000**
CEP (FPR)	0.7653	0.1637	**0.0002**	0.0001
FairTop-k	0.1834	0.1639	0.0038	0.0001
Conference Decision	0.1851	**0.0040**	0.0009	0.0009

maximizing fairness perception and minimizing the weighted statistical disparity of the different groups.

5.6 Conclusions

This chapter presents a novel fairness-aware peer review recommendation algorithm called FPRank that considers a multi-objective criteria based on acceptability scores, fairness perception, and weighted statistical disparity, to generate its recommendation for paper acceptance. We also provide theoretical analysis to show that maximizing fairness perception alone is insufficient as it may lead to bias in terms of the group fairness criteria. Finally, we successfully demonstrate the effectiveness of FPRank in terms of balancing the competing requirements using open review data from a major conference in machine learning.

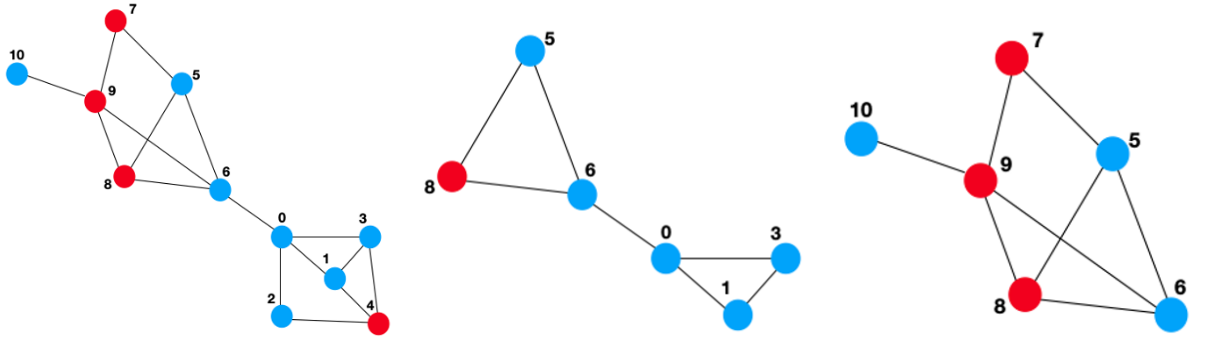
CHAPTER 6

FAIR GRAPH SAMPLING

This chapter investigates the problem of network sampling from a fairness perspective. First we propose a novel fairness-aware network sampling framework that combines the structural preservability and group representativity objectives into a unified structure. Next we developed a fair greedy sampling algorithm which aims to satisfies both objectives. We also provide a theoretical approximation guarantee bound for the proposed method.

6.1 Introduction

Networks are powerful tools for modeling interactions between entities in a complex social system. The digital revolution has provided a unique opportunity to collect significant amount of information related to such systems. For example, the number of smartphone users has grown by 40% from 2016 to 2021, reaching up to 3.8 billion people [163]. The pervasiveness of such technology means more social network users, more online purchases, more user-generated content, and consequently, more massive networks. Executing even simple algorithms on such a massive network can be very expensive. In addition to computational complexity, many online social networks are not entirely visible due to privacy concerns, and are accessible only via crawling or



(a) The Original Network (b) An Structural Preservative Sam-(c) A Group Representative Sample ple

Figure 6.1: An example illustration of different sampling objective.

through the use of APIs. However, such access is often limited due to restrictions on the number of queries or rate limits imposed by the data providers [1]. This makes accessing the whole network extensively hard or near impossible.

One way to address the above challenges is by using network sampling. Sampling can be used to extract a subset of the nodes and links of the network for subsequent analysis tasks. For example, the sampled network can be used for efficient training of sophisticated deep learning models such as graph convolutional networks (GCN) or for network visualization purposes. Furthermore, it can be utilized as an efficient way to collect data with a limited budget.

The most common sampling objective is to preserve specific topological properties of the original network, which will be referred to as *structural preservability* in this chapter. In graph sampling literature, many network topological measures have been used to characterize structural preservability. These properties can be generally categorized into two groups: (1) vector properties such as the distribution of node degrees, clustering coefficients, and eigenvalues, and (2) scalar properties such as average degree, network diameter, and average clustering coefficient. The former can be assessed using probabilistic distance measures such as Kullback–Leibler divergence and Kolmogorov-Smirnov statistic whereas the latter can be measured using measures such as normalized root mean square error (NRMSE) [73].

Besides their link structure, many networks can also be characterized by properties of their node attributes. The sampling objective can therefore be expressed in terms of representativeness of their node attribute values [73], which will be referred to herein as *group representativity*. The node attributes used to define the groups are known as protected attributes (e.g., gender, race, age group, etc). Group representativity can be formalized as the ratio of nodes from each subgroup in the sampled subgraph to the original network. Figure 6.1 shows a toy example to illustrate the two potentially conflicting sampling objectives. The network in Figure 6.1(a) has 11 nodes with the red and blue colors representing their group memberships. Observe that there are two major clusters in the network, one involving the nodes labeled 0-4 while the other involving nodes 5-10. Figure 6.1(b) illustrates a structurally preservable network sample that maintains the clustering structure but is not group representative as it includes mostly nodes from the blue subgroup, with only one node from the red subgroup. On the other hand, Figure 6.1(c) illustrates a group representative sample that is not structurally preservable since it contains nodes primarily from one of the two clusters.

Similar to other machine learning tasks, existing network sampling algorithms are not impervious to biases against certain demographic groups of the social network. Given its broad range of applications, the importance of fair network sampling cannot be overly emphasized. If the sampled network is biased, this will adversely affect the results of downstream mining tasks. For example, Wagner et al. [166] showed that uninformed sampling may lead to biased estimation of node centrality values and unfair ranking of nodes from minority groups in a social network. They also proposed a metric based on the idea that an ideal sample should not “systematically rank nodes of one group higher and nodes of the other group lower than expected." However, the authors did not present any new sampling method to overcome the limitations of existing algorithms.

The main challenge in fair network sampling is how to combine the structural preservability and group representativity objectives in a principled way and to design an algorithm that optimizes for both. In this chapter, we develop an approach that measures structural preservability by comparing the centrality measures of the nodes in the sampled network compared to their values in the original network. The fairness of the sampled network, which corresponds to our group representativity measure, is then defined by a max-min subgroup fairness criterion [118] based on the worst-case structural preservability value among all the subgroups of the protected attribute. We have also developed a greedy algorithm to obtain an approximate solution for the max-min subgroup fairness criterion. A systematic evaluation of the proposed sampling algorithm was performed on various real-world social network data to empirically compare its effectiveness relative to other baseline methods.

6.2 Related Work

Previous methods for network sampling can be categorized as random node selection (RN), random edge selection (RE), or traversal-based sampling methods. In the first case, the RN sampling algorithm will select a subset of the nodes, while maintaining all the edges among the selected nodes. The sampled nodes are selected according to some structural preservability criteria, e.g., in terms of their node degree, PageRank value [138], or other node centrality measures. In the second case, the sampling algorithm will select a subset of the edges to create the subgraph. The RE sampling process can be done by randomly choosing the edge or by randomly choosing a node first before selecting an edge incident to the node [104]. Finally, a traversal-based sampling algorithm starts from an initial set of nodes (and/ or edges) and expands the sample based on their connectivity to the nodes in the current sample. This method includes random walk based sampling [174, 122, 148], snowball sampling [57] (which is similar to breadth-first search with the difference in that it only expands a fixed number of neighbors), forest fire sampling [105] (which is a probabilistic version of snowball sampling), and Markov chain Monte Carlo methods (e.g., the Metropolis-Hastings [75, 61] algorithm).

Graph sampling methods can be utilised in training graph neural networks (GNN). For example, Hamilton et al. [67] proposed a method for generating node representation by sampling and aggregating features from a node's local neighborhood. Ying et al. [183] combines random walks and graph convolutions to generate node representations. Chen et al.[28] proposed a sampling scheme in the reformulation of the loss and the gradient whereas Zou et al.[194] proposed layer-dependent importance sampling. Alternative graph sampling methods have been proposed in [31, 187, 35]. These sampling approaches are designed for efficient training of graph neural networks, instead of preserving the topological properties and group representativity of the network, which is the focus of this chapter.

Fairness in machine learning has received considerable attention in recent years [16, 49, 69, 97]. Prior works have mainly focused on classification tasks and can be categorized into individual and group-level fairness. Individual-level fairness was developed based on the proposition that similar people should be treated similarly. An example of such an approach is by Dwork et al. [49] that defines a metric based on probabilistic distance measure between individuals with Lipschitz condition. Fairness from a group-level perspective was developed based on the idea that members of different demographic groups such as race and gender should be treated equally. These approaches would quantify fairness in terms of statistical measures [16] such as statistical parity [49], equalized odds [69], and balanced error rate [53].

There are growing interests to the problem of fairness in network mining tasks [121, 147, 23, 81, 40]. Rahman et al. [147] proposed Fairwalk, a fairness-aware embedding method based on node2vec. Buyl and De Bie [23] proposed a Bayesian method for learning debiased embeddings by using a biased prior. Kanget al. [81] presented a definition of individual fairness for graph mining and three algorithms for debiasing the input graph, debiasing the learning model and debiasing the output results. Dai and Wang [40] proposed debiasing framework for GNNs whereas Masrour et al. [121] introduced a fairness metric based on a network modularity measure and utilized it for link prediction tasks in network mining to alleviate the filter bubble problem in social networks.

6.3 Preliminaries

Let $G =< V, E, X >$ be an attributed network, where V is the set of nodes, $E \subseteq V \times V$ is the set of links, and $X \in \mathbb{R}^{|V| \times d}$ is the feature matrix associated with the nodes in the network. We assume X can be partitioned into two groups $X = (X^{(p)}, X^{(u)})$, where $X^{(p)}$ denote the protected attributes and $X^{(u)}$ denote the unprotected ones. Let $\{P_1, \ldots, P_K\}$ be the partitions of the nodes in V based on the values of the protected attribute $X^{(p)}$, where K is the number of distinct combination of its values.

Given a pair of networks, $G_1 =< V_1, E_1, X_1 >$ and $G_2 =< V_2, E_2, X_2 >$, we say that G_1 is a subgraph of G_2, denoted as $G_1 \subseteq G_2$, if $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$. If E_1 includes all the links in G_2 that have endpoints in V_2, then G_1 is an induced subgraph of G_2. We denote the set of all possible induced subgraphs of size n of a graph G as $\mathbb{G}_n$. For brevity, we consider only undirected networks in this study, though the ideas can be extended to directed networks.

Definition 12 (Network Sampling)**.** *Given an attributed network $G =< V, E, X >$ and a sample size n, the task of network sampling is to extract an induced subgraph $G_s^* =< V_s^*, E_s^*, X_s^* >$ with n nodes that maximizes the following objective function $\mathbf{o} : \mathbb{G}_n \rightarrow \mathbb{R}^+$.*

$$G_s^* = argmax_{G_s \in \mathbb{G}_n} \mathbf{o}(G_s) \tag{6.1}$$

This chapter focuses on designing a network sampling algorithm that satisfies both structural preservability and group representativity requirements. The former requires comparing the topological properties of the sampled graph against the original network while the latter evaluates how representative the sample is with respect to different subgroups, as defined by the node attributes. We will discuss in detail the proposed objective function to achieve this goal in the next section.

6.3.1 Network Sampling as Subset Selection Problem

The network sampling problem can be formulated as a subset selection problem, which is defined as the task of finding a subset A from a set S that maximizes an objective function $f : 2^S \rightarrow \mathbb{R}^+$ such that $|A| \leq n$. Network sampling is a form of subset selection problem where S is the set of nodes in the network. Besides network sampling, there are other machine learning tasks, such as feature subset selection, that can also be formulated as a subset selection problem. However, due to its exponential search space, it has been shown that the subset selection problem is usually NP-hard[43].

Many algorithms have been developed to find a polynomial-time approximation solution for this problem. These algorithms can be generally categorized into two groups—convex relaxation and greedy algorithms [146]. Convex relaxation methods would replace the non-convex subset size constraint with convex constraints [161, 195]. In contrast, greedy algorithms would start from an empty set and iteratively select an element from the set V to be added in a way that maximizes the objective function [60, 43]. For submodular objective functions, the greedy algorithms are guaranteed to find a reasonably good approximation to the optimal solution. Specifically, a function

f is submodular if and only if:

$$f(A \cup B) + f(A \cap B) \leq f(A) + f(B) \tag{6.2}$$

for all subsets $A, B \subseteq V$. If f is a monotone, submodular, non-negative function on 2^V, then the greedy algorithm will provide a solution within $1 - \frac{1}{e}$ approximation of the optimum solution [129].

If the function does not have submodular property, the greedy algorithms may perform poorly. Nevertheless, there is a class of non-submodular functions that has received considerable attention recently. Bian et al. [19] provided an approximation of the bounds for greedy maximization of nondecreasing set functions. Their bound was defined in terms of the (generalized) curvature and submodularity ratio of the function. Chen et al. [29] showed that a randomized version of the greedy algorithm achieves an approximation ratio of $(1+\frac{1}{\gamma})^{-2}$ for weakly submodular maximization subject to a general matroid constraint, where γ is a parameter measuring the distance from submodularity.

6.4 Proposed Sampling Framework

Our goal is to design a network sampling framework that considers both structural preservability and group representativity objectives. In this section we first introduced the proposed node-level sampling measure for structural preservability of the sample. We then describe our group representativity objective, which measures fairness in the sampled network in terms of the worst-case structural preservability measure for different groups of the protected attribute. The criteria will be employed by our proposed greedy algorithm to be described in Section 6.5.

6.4.1 Node-level Sampling Measure

Since the protected attribute is an individual-level feature defined for each node, a natural starting point for measuring a sample's goodness in terms of preserving the topological properties of a network is at the individual node level. More specifically, node centrality [130] has been widely used as a measure of the importance of the nodes in a network. By comparing the centrality values of all the nodes in the sampled graph to their corresponding values in the original network, the structural preservability of a sample can therefore be evaluated.

Definition 13 (Centrality Ratio)**.** *Given a network $G =< V, E, X >$ and a subgraph G_s, then the centrality ratio μ for a node v in G_s is*

$$\mu(v, G_s) = \frac{C(v, G_s)}{C(v, G)} \tag{6.3}$$

where C corresponds to a node centrality measure.

One potential caveat of using Equation (6.3) is that it assumes the entire network is available in order to compute its denominator. If the sampling algorithm is restricted to have access only to the sampled network (e.g., while crawling the network) instead of full access to the entire network, we may relax the preceding definition to the following equation instead:

$$\tilde{\mu}(v, G_s) \approx C(v, G_s) \tag{6.4}$$

The relaxed measure is often used during sampling by most algorithms whereas the true centrality ratio (Equation (6.3)) is used when evaluating the performance of the sampling algorithm. Note that there are numerous centrality measures that have been introduced in the literature [130]. In this chapter, we consider the harmonic centrality [22] as our node centrality measure. Harmonic centrality of a node u to a graph G is defined as the sum of the inverse of the shortest path distance, $d_G(u, x)$, between u to all other nodes x in G:

$$H(u, G) \equiv \sum_{x \in V \setminus \{u\}} H(u, x) \ = \sum_{x \in V \setminus \{u\}} \frac{1}{d_G(u, x)} \tag{6.5}$$

Note that although the discussion in this chapter focuses on harmonic centrality, our proposed framework is applicable to other centrality measures as well. We choose harmonic centrality as our measure for several reasons. First, unlike other centrality measures such as closeness and betweenness, which are restricted to connected networks, harmonic distance is applicable to both connected and disconnected networks. This is because if $d_G(u, x) = \infty$, then the corresponding element in the sum given in Equation (6.5) will be zero. Also, the measure is intuitive as it considers the relative influence of the nodes in a network by giving higher weights to nodes that are closer to u than those located further away. The strength of using harmonic distance compared to other centrality measures was recently demonstrated in [22], where the authors presented an axiomatic approach to evaluate the various centrality measures according to a set of fundamental properties a good centrality measure should exhibit. They showed that only harmonic centrality measure satisfies all the axioms, which makes it an excellent measure for arbitrary networks [21]. For example, the authors showed that harmonic centrality is one of only two centrality measures (besides PageRank) that is strictly rank monotone, i.e., adding a new edge to a node will not demote its rank relative to other lower-ranked nodes in the network.

Furthermore, the following lemma shows two additional properties of the harmonic centrality when used as he structural preservability measure (see Equation (6.4) by a sampling algorithm.

Lemma 6. *The harmonic centrality satisfies the following properties:*

1. $\tilde{\mu}(v, G_s) = 0$, *if* $v \notin V_s$.

2. $\tilde{\mu}(v, G_1) \leq \tilde{\mu}(v, G_2)\ \forall v \in V_1$, *if* $G_1 \subseteq G_2$.

where $\tilde{\mu}(v, G_s) = H(v, G_s)$.

Proof: For the first property, it is easy to show that $\tilde{\mu}(v, G_s) = 0$ for $v \notin V_s$ since $\forall u \in V_s$: $d_G(v, u) = \infty \Rightarrow H(v, u) = 0$. To prove the second property, let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. If $G_1 \subseteq G_2$ then $\forall v, u \in V_1 : d_{G_1}(v, u) \geq d_{G_2}(v, u)$. As a result,

$$\forall v, u \in V_1 : \frac{1}{d_{G_1}(v, u)} \leq \frac{1}{d_{G_2}(v, u)}.$$

Since $V_1 \subseteq V_2$, it is easy to show that $H(v, G_1) \leq H(v, G_2)$, $\forall v \in V_1$. Thus, the second property holds for Equation (6.4). □

6.4.2 Fairness-aware Network Sampling Objective

To address the challenge of combining structural preservability with group representativity, we introduce a max-min subgraph fairness criterion, which is inspired by the idea of the minimax Pareto fairness concept proposed in [118] for satisfying group fairness. Specifically, our max-min subgraph fairness criterion evaluates the average centrality value for each group of the protected attribute and uses its worse group performance (i.e., group with minimum average centrality value) as its fairness measure.

Let $\{P_1, P_2, \cdots, P_K\}$ be a partitioning of the nodes in a network G based on their membership according to the protected attribute $X^{(p)}$. For each group P_i, we define the following group centrality ratio as follows:

Definition 14 (Group Centrality Ratio)**.** *Given a network $G =< V, E, X >$ and centrality ratio μ, the group centrality ratio for the node group P_i is*

$$\sigma_i(G_s) = \frac{1}{|P_i|} \sum_{u \in P_i} \mu(u, G_s) \tag{6.6}$$

For sampling algorithms without full access to the entire network, we may replace μ by $\tilde{\mu}$ in the above definition.

Definition 15 (Subgraph Fairness Criterion)**.** *Given a network sample $G_s =< V_s, E_s, X_s >$, node groups $\{P_1, P_2, \cdots, P_K\}$, and a group centrality ratio function σ, we define the following subgraph fairness measure:*

$$\mathbf{o}_{fair}(G_s) = \min_{1 \leq i \leq K} \{\sigma_i(G_s)\} \tag{6.7}$$

Our network sampling goal is to maximize the fairness criterion defined in Equation (6.7). Replacing the measure into the objective function in Equation (6.1), our fairness-aware sampling objective is:

$$G_s^* = \text{argmax}_{G_s \in \mathbb{G}_n} \min_i(\sigma_i(G_s)) \tag{6.8}$$

Lemma 7. *If $\tilde{\mu}$ satisfies the properties stated in Lemma 6, then the fair network sampling measure, $\mathbf{o}_{fair}$, is a monotonically nondecreasing function.*

Proof: Let $\sigma^{min}(G_s) = \min_{1 \leq i \leq K}(\sigma_i^T)$. In order to show σ^{min} is a monotonically nondecreasing function, we have to show that $\sigma^{min}(G_1) \leq \sigma^{min}(G_2)$ if $G_1 \subseteq G_2$. Since $\tilde{\mu}$ satisfies the second property in Lemma 6 and σ_i^T is a summation over μ, therefore σ_i^T must be a monotonically nondecreasing function for all $1 \leq i \leq K$. Furthermore, as σ^{min} is the minimum value of σ_i^T over i, it must also be monotonically nondecreasing, which completes the proof. □

6.5 Proposed Greedy Fair Network Sampling Algorithm

This section describes our proposed greedy sampling algorithm for fair network sampling, which is designed to provide an approximate solution to the sampling objective given in (6.8).

6.5.1 Greedy Algorithm for Max-min Subgraph Fairness

We propose the following greedy algorithm for the max-min subgraph fairness problem. Our greedy algorithm is based on the following notion of marginal gain of a set of nodes $A \subset V$.

Definition 16 (Marginal Gain)**.** *Given a network $G = (V, E)$, a subgraph $G_s = (V_s, E_s)$, and a set of nodes $A \subset V \setminus V_s$, the marginal gain $\delta^{min}(.)$ of adding set $A \subset V$ to G_s is*

$$\delta^{min}(A|G_s) = \sigma^{min}(G_s \cup A) - \sigma^{min}(G_s) \tag{6.9}$$

where $\sigma^{min}(G_s) = \min_{1 \leq i \leq K}\{\sigma_i^T(G_s)\}$.

A greedy algorithm can be developed to optimize (6.7) by incrementally adding a node v into the sample G_s in a way that maximizes the marginal gain. However, computing the harmonic centrality can be expensive when the sampled graph is large. To improve its efficiency, we present the following fast implementation of our greedy algorithm based on a reference set of target nodes.

Definition 17 (Target Set)**.** *Given a network $G =< V, E, X >$, where $V = \cup_{i=1}^K P_i$, the target set $T = \{T_1, \ldots, T_k\}$ is a set of node subsets such that $T_i \subset P_i$, for $1 \leq i \leq K$.*

We will use the target set to compute the following approximate group centrality ratio: $\tilde{\sigma}_i(G_s) = \frac{1}{|T_i|} \sum_{u \in T_i} \mu(u, G_s)$ for each group and compute the marginal gain in Equation (6.9) using the approximate group centrality ratio instead. Thus, our greedy algorithm will start from the initial, induced subgraph of the target set T, and then, iteratively expands the subgraph by selecting a node from the candidate set $\chi(G_s)$ that maximizes the marginal gain. The candidate set $\chi(G_s)$

corresponds to all immediate neighbors of the nodes V_s in $V \setminus V_s$. Note that the target set T may correspond to the initial seed nodes of the sampling algorithm or a subset of nodes that must be included in the network sample (e.g., nodes with high degrees or other important nodes as specified by user).

Algorithm 1 Greedy Fair Network Sampling (GFNS)

```
Input: graph G, sample size n, and target set T.
$G_0 \leftarrow$ Induced-subgraph(T).
for t = 1 to $n - 1$ do
    $\chi \leftarrow \{u \mid (u, v) \in G, u \in V \setminus V_{t-1}, v \in V_{t-1}\}$
    $v^* \leftarrow \text{argmax}_{v \in \chi} \delta^{min}(v|G_{t-1})$
    $G_t \leftarrow$ Induced-subgraph($G_{t-1} \cup \{v^*\}$)
end for
Output: $G_t$
```

6.5.2 Theoretical Bounds on Greedy Approximation

Unfortunately, the min-max subgraph fairness criterion is not a submodular function. However, as shown in Lemma 7, it is a monotonically nondecreasing function. This allows us to use the result of [19] to obtain a theoretical bound on the greedy approximated solution. Before providing the main theorem, we first need to introduce some definitions.

Definition 18 (Greedy Submodularity Ratio [19])**.** *The greedy submodularity ratio of a function σ is the largest scalar γ such that*

$$\sum_{u \in A \setminus G^{(t)}} \delta(u|G^{(t)}) \geq \gamma \delta(A|G^{(t)}), \quad \forall |A| = n, t = 0, \ldots n - 1$$

Remark. *For a non-decreasing σ or σ^{min} functions, $\gamma \in [0, 1]$.*

Definition 19 (Greedy curvature [19])**.** *The greedy curvature is the smallest scalar α*

$$\delta(v_t|G^{(t-1)} \cup A) \geq (1 - \alpha)\delta(v_t|G^t), \forall |A| = n$$

Theorem 8. *Let σ be the group centrality measure defined in Equation* (6.6) *and $\delta^{min}(\cdot)$ be the marginal gain with greedy submodularity ratio and greedy curvature defined in Definitions 18 and 19, respectively. The proposed greedy fair network sampling algorithm has the following approximation guarantee*

$$\sigma(T, G^{(K)}) \geq \frac{1}{\alpha}\Big[1 - \Big(\frac{K - \alpha\gamma}{K}\Big)^K\Big]\sigma(T, G^*) \geq \frac{1}{\alpha}(1 - e^{-\alpha\gamma})\sigma(T, G^*)$$

where $G^{(K)}$ is the output of the greedy algorithm and G^* is the optimum solution.

The proof of the theorem can be shown using Lemma 7 of this chapter and Theorem 1 of [19].

6.6 Experimental Evaluation

This section describes the experiments performed to evaluate the efficacy of our proposed methods fair network sampling algorithm.

6.6.1 Experiment Setup

6.6.1.1 Datasets

We evaluated our methods on four real-world datasets as summarized in Table 6.1. The first dataset corresponds to the Facebook ego-network of friendship relation [106]. The second dataset corresponds to a social network from `tagged.com` [52]. The third dataset is the German credit[48] data, where the nodes represent clients and links between nodes are created based on similarity of the clients' credit accounts [5]. Finally, the fourth dataset corresponds to the Credit Default data [182], where the nodes are credit applicants and the links are created based on similarity of the applicants [5]. Gender is chosen as the protected attribute for the first 3 datasets whereas age is the protected attribute for the last dataset.

6.6.1.2 Evaluation Metric

We evaluate the structural preservability of the sampling algorithm according to the following three metrics:

Table 6.1: Statistics of network data used for experiments.

Network	#nodes	#edges	CC	protected feature
Facebook	4,039	88234	0.6055	gender
Tagged	5,607,448	912,280,409	0.0005	gender
German	1,000	24,970	0.3801	gender
Credit	30,000	2,174,014	0.6466	Age

- **Degree distribution distance (Ddist):** Using the Kolmogorov-Smirnov statistic, we compare degree distribution of the sampled network to that of the original network. It is a nonparametric test of equality between continuous, one-dimensional probability distributions. Following [104], we simply use it to measure the distance between the two distributions as follow:

  $$Ddist(G_s) = \sup_d |F_S(d) - F(d)|$$

  where F_S and F are cumulative distribution functions (CDFs) of degree distributions for the sampled network and original network respectively.

- **Clustering Coefficient(δ-CC):** We compare the average clustering coefficient of the sampled graph and the original one.

  $$\delta\text{-CC} = \left|\frac{1}{|V|}\sum_{v\in V}\frac{\lambda_G(v)}{\tau_G(v)} - \frac{1}{|V_s|}\sum_{v\in V_s}\frac{\lambda_{G_S}(v)}{\tau_{G_S}(v)}\right|$$

  where $\lambda_G(v)$ is the number of triangles in G with v being one of the nodes. $\tau_G(v)$ is the number of subgraphs in G with 2 links and 3 nodes, where one of the nodes is v and v is connected to the other two nodes.

- Harmonic: the average centrality ratio efined based on Harmonic distance for all nodes in the sampled graph. This measure will be between zero and one. Larger values are better.

For group representativity, we consider the following evaluation metrics:

- **Normalized Cumulative Group Relevance (nCGR)** [166]: measures the extent to which the position of the nodes in the ranking of a sampled graph and the visibility of a protected group is preserved. To do this, the relevance of a node v in a given graph is defined as the inverse of the node v's rank, based on its degree, normalized by the rank sum of all nodes in the network.

$$rel(v) = \frac{(rank(v))^{-1}}{\sum_{u \in V} rank(v)}$$

The cumulative protected group relevance for the sampled graph is compared against the original network as follows:

$$nCGR_i = \frac{\sum_{v \in topk(G_s) \cap P_i} rel(v) + \epsilon}{\sum_{v \in topk(G) \cap P_i} rel(v) + \epsilon}$$

where $nCGR_i$ measures the extent to which the relevance of a protected group P_i is above or below the expectation from the original network with respect to the top k nodes. Value above 1 indicates the group is more relevant in the sampled graph compared to the original network. Value less than 1 indicates that the group has become less relevant in the sampled network. The hyperparameter $\epsilon = 0.001$ is used to avoid division by zero [166]. Here we report the minority group nCGR and the closer value to 1 is considered to be better.

- **min-σ**: The subgraph fairness criterion using the group centrality ratio defined in Equation (6.6). Here in evaluation we consider all the nodes in the sampled subgraph when computing σ.

6.6.1.3 Baseline Sampling Algorithms

We consider following commonly studied sampling techniques

- Random Node: We consider two variations; the first variation which we refer to as **NS**, randomly selects a subset S of the nodes without considering their topology properties. The second variation, **NSD**, randomly selects a subset S of the nodes with probability proportional to the node degree.

- Breadth/Depth-first search **BFS/ DFS**: Both BFS and DFS are widely used for exploring large networks. Both algorithms start from an initial set of nodes and iteratively expand the sample based on their graph traversal strategy.
- Random Walk: **RW** starts with an initial set of seed nodes and expands the sample by simulating a random walk on the network. Fair Random Walk (**FRW**) [147] is a variation of the method to account for fairness. Instead of randomly selecting a node amongst all neighbors as the next node, it first partitions the neighbors into groups based on their protected attributes such that each group has the same probability of being chosen regardless of their sizes. A random node from the chosen group is then selected as the next node to visit.
- Metropolis-Hastings Random Walk (**MHRW**): Metropolis-Hastings is a Markov Chain Monte-Carlo (MCMC) technique for producing random samples from an arbitrary distribution. The MHRW results in selecting a subset S of nodes from a uniformly random distribution.

In all of the above methods, after selecting a subset S of nodes, we extract the induced subgraph corresponding to the nodes in S.

6.6.2 Experimental Results

In the following subsections we investigate the performance of the proposed framework.

6.6.2.1 General Performance

In this subsection we investigate the performance of the proposed sampling methods on preserving network properties. Tables 6.2, 6.3, 6.4, and 6.5 shows the result on all seven measures for the four datasets. Based on the results, there is no single approach which perform better than other sampling methods on all 4 datasets. Specifically, for the three structural perservability related measures the GFNS perform relatively better than other baselines. Although the results may vary depending on the dataset and evaluation metric the GFNS algorithm consistently appears in the top-2 in 9

Table 6.2: German credit Dataset

	δ-CC	Ddist	Harmonic	nCGR	min-σ
NS	**0.008+/-0.00**	0.616+/-0.00	0.368+/-0.00	1.01+/-0.00	0.365+/-0.00
NSD	0.009+/-0.00	0.412+/-0.00	0.4 +/-0.00	1.018+/-0.00	0.398+/-0.00
DFS	0.008+/-0.00	0.586+/-0.00	0.372+/-0.00	1.011+/-0.00	0.369+/-0.00
BFS	0.027+/-0.000	0.333+/-0.00	0.409+/-0.00	1.018+/-0.00	0.408+/-0.00
RW	0.016+/-0.00	0.351+/-0.001	0.408+/-0.00	1.014+/-0.00	0.407+/-0.00
MHRW	0.008+/-0.00	0.355+/-0.001	0.405+/-0.00	**1.008+/-0.00**	0.401+/-0.00
FRW	0.011+/-0.00	0.357+/-0.001	0.406+/-0.00	1.014+/-0.00	0.405+/-0.00
GFNS	0.037+/-0.00	**0.249+/-0.00**	**0.443+/-0.00**	1.019+/-0.00	**0.441+/-0.00**

Table 6.3: Faecebook Dataset

	δ-CC	Ddist	Harmonic	nCGR	min-σ
NS	0.177+/-0.002	0.659+/-0.001	0.015+/-0.00	1.039+/-0.00	0.014+/-0.00
NSD	0.077+/-0.00	0.224+/-0.001	0.079+/-0.00	1.018+/-0.00	0.078+/-0.00
DFS	0.02+/-0.00	0.538+/-0.00	0.024+/-0.00	1.025+/-0.00	0.023+/-0.00
BFS	0.105+/-0.00	0.308+/-0.00	0.109+/-0.00	**1.007+/-0.00**	0.109+/-0.00
RW	0.103+/-0.00	0.339+/-0.001	0.093+/-0.00	1.012+/-0.00	0.092+/-0.00
MHRW	0.041+/-0.00	0.192+/-0.005	**0.145+/-0.00**	1.013+/-0.00	**0.144+/-0.00**
FRW	0.096+/-0.00	0.341+/-0.002	0.089+/-0.00	1.012+/-0.00	0.087+/-0.00
GFNS	**0.014+/-0.00**	**0.182+/-0.00**	0.135+/-0.00	1.009+/-0.00	0.135+/-0.00

out of 12 settings, which shows the effectivness of the proposed method in preserving structural properties of the graph.

For the sampling fairness evaluation, we consider the proposed criteria alongside nCGR as the evaluation metrics. The last two columns in Tables 6.2, 6.3, 6.4, and 6.5 results. Similar to structural preservability, The results indicate that no single approach performs better than other sampling methods on all four datasets. The proposed greedy algorithm successfully achieves to be among the top-2 in 6 out of 8 settings. On the Credit dataset, the performance of our algorithm on both fairness criteria is the best. On the other three datasets, it has the best result in one of the two fairness criteria.

6.6.2.2 Performance on Target Nodes

We first evaluate the performance of the sampling algorithms on target nodes. We ran each baseline algorithm ten times, and plot the patterns. For the target set, we consider ten nodes, ten nodes

Table 6.4: Credit Dataset

	δ-CC	Ddist	Harmonic	nCGR	min-σ
NS	0.523+/-0.006	0.999+/-0.00	0.001+/-0.00	1.012+/-0.00	0.000+/-0.00
NSD	0.242+/-0.007	0.997+/-0.00	0.001+/-0.00	**1.000+/-0.00**	0.001+/-0.00
DFS	0.146+/-0.00	0.984+/-0.00	0.006+/-0.00	**1.000+/-0.00**	0.006+/-0.00
BFS	0.121+/-0.00	0.868+/-0.00	0.008+/-0.00	1.051+/-0.00	0.007+/-0.00
RW	**0.019+/-0.00**	0.808+/-0.00	0.007+/-0.00	1.017+/-0.00	0.007+/-0.00
MHRW	0.081+/-0.002	0.552+/-0.018	0.016+/-0.00	1.001+/-0.00	0.012+/-0.00
FRW	0.033+/-0.00	0.768+/-0.002	0.007+/-0.00	1.004+/-0.00	0.006+/-0.00
GFNS	0.145+/-0.00	**0.182+/-0.00**	**0.026+/-0.00**	**1.000+/-0.00**	**0.015+/-0.00**

Table 6.5: Tagged Dataset

	δ-CC	Ddist	Harmonic	nCGR	min-σ
NS	0.001+/-0.00	0.398+/-0.00	**0.217+/-0.12**	2.052+/-0.00	**0.208+/-0.12**
NSD	0.001+/-0.00	0.105+/-0.003	0.012+/-0.001	1.951+/-0.00	0.01+/-0.00
DFS	0.001+/-0.00	0.298+/-0.00	0.026+/-0.00	1.048+/-0.00	0.107+/-0.00
BFS	0.001+/-0.00	0.193+/-0.00	0.115+/-0.00	1.072+/-0.00	0.113+/-0.00
RW	0.001+/-0.00	0.30+/-0.00	0.114+/-0.00	1.053 +/-0.00	0.106+/-0.00
MHRW	0.002+/-0.00	0.267+/-0.005	0.015+/-0.00	1.067+/-0.00	0.012+/-0.00
FRW	0.001+/-0.00	0.281+/-0.00	0.114+/-0.00	1.052+/-0.00	0.111+/-0.00
GFNS	0.001+/-0.00	**0.082+/-0.00**	0.079+/-0.00	**1.040+/-0.00**	0.068+/-0.00

from each gender. For the Facebook data, we consider nodes with the highest harmonic centrality measure as the target set. For the Tagged network, we randomly selected five nodes from each gender.

Figure 6.2 shows the value of μ for the ten target nodes on the Facebook dataset for the sample size of 5% of the original network. Here we sorted the value of μ for each sampling method. We notice that the GFNS outperform all the baseline method. For all ten nodes in the target set, the value of μ defined based on Harmonic distance is higher for GFNS. This is what we expected, and the experiment results agree with it. The second observation is that the BFS is performing better than most of the baselines.

Table 6.6 for German credit network summarizes the performance of the algorithms based on the average of proposed measure in equation 6.7 for different sample sizes. The higher value indicates better performance in preserving the structural properties of target nodes. It is transparent that regardless of the sample size and network type, the proposed greedy algorithm outperforms

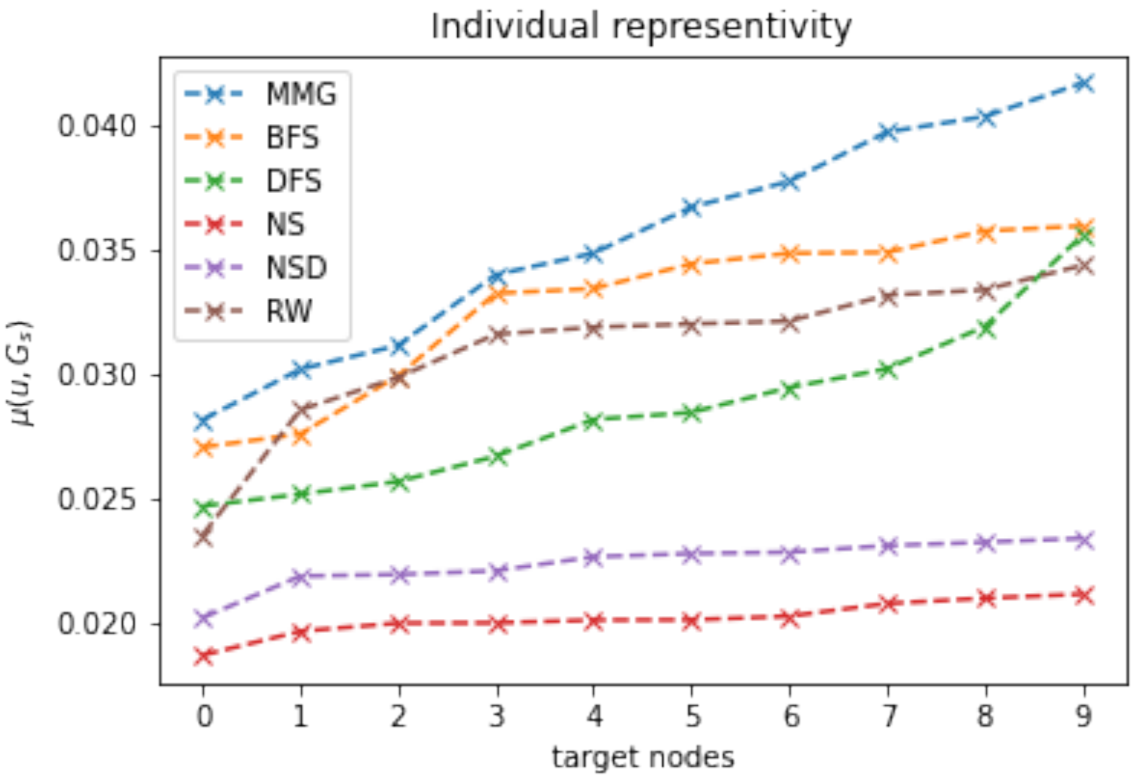


Figure 6.2: The proposed Harmonic measure values for the samples with 10 target nodes.

Table 6.6: The average node-level harmonic measure for German credit network .

	100	200	400
NS	0.0617+/-0.0001	0.1629+/-0.0001	0.3639+/-0.0001
NSD	0.0841+/-0.0000	0.1900+/-0.0000	0.3984+/-0.0000
DFS	0.0765+/-0.0000	0.1651+/-0.0000	0.3672+/-0.0000
BFS	0.0985+/-0.0000	0.2033+/-0.0000	0.4211+/-0.0000
RW	0.0850+/-0.0000	0.2013+/-0.0000	0.4129+/-0.0000
FRW	0.0870+/-0.0000	0.1984+/-0.0000	0.4114+/-0.0000
GFNS	**0.1147+/-0.0000**	**0.2259+/-0.0000**	**0.4396+/-0.0000**

other sampling methods as expected. This result shows the proposed greedy sampling is successful in achieving its objective.

6.7 Conclusion

This work presents a novel fairness-aware network sampling approach that combines the structural preservability and group representativity objectives into a unified learning framework. We introduced a new sugraph fairness criterion and developed a greedy fair network sampling algorithm with well-grounded theoretical bounds on the greedy approximation. Finally, we experimentally demonstrate the effectiveness of the proposed method on various real-world network data.

CHAPTER 7

FUTURE WORK

The use of machine learning algorithms in daily life is pervasive and entrenched in our society. Its vast application includes different fields such as healthcare, criminal justice, advertisement and recommender systems, banking and finance, dating and hiring, online social media. This broad application creates a unique opportunity for machine learning researchers with new data sources, problems, and challenges to address. Machine learning and data analysis can help us expand our knowledge about the complex environment we live in and makes it more accessible and efficient. It also exposed to the social, cultural, and institutional biases that have been established in our society.

I want to continue my research in two directions. First, I would like to extend my research in network analysis. Although network studies have played a central role in machine learning for quite a while, there are still multiple challenges in this field. In the first part of this chapter, I explain some of the related challenges that I would like to address in my research. Second, I would like to expand my line research in fairness to explainability. During my research related to this thesis, I realized that explainability and fairness are related, and explainability can play a significant role in mitigating machine learning-generated biases. In the second part of this chapter, I will explain this connection and some research ideas that I would like to investigate.

7.1 Network Analysis

I would like to extend my line of research on fairness-aware network analysis in the near future. Many other problems in fairness network analysis are non-i.i.d problems and therefore call for an approach similar to the ones that I have developed in my previous work. An interesting question I would like to investigate is whether the network fairness measures that I have proposed can also be applied to dynamic networks. This is not a straightforward question considering the fact that static approach network centrality measures, such as PageRank, might be inappropriate in dynamic settings [117]. An example of a fairness-aware dynamic problem on which I would like to work is the ranking node problem. This has essential applications in recommender systems and information retrieval. While there is some recent work on fairness in the non-dynamic ranking node problem [176], to the best of my knowledge, there is no work on the dynamic ranking node problem.

In addition to the fairness-aware network research track, I also found expanding the computational power of current network AI models beyond the message passing framework a critical problem that I would like to investigate. I am interested in developing new network learning algorithms that incorporate higher-order structural information about networks into the AI model. I believe the harmonic-based measure for the sampling with an application on training graph convolution networks has connections to the notion of graph moments discussed in [9] that I would like to study more.

7.2 Fairness and Explainability in Machine Learning

The best way to quantify fairness remains an unresolved debates among experts. There are strong criticisms against both individualistic and group level approaches for quantifying fairness. For example, group level metrics fail to guarantee fairness for individuals and subgroups, whereas individual fairness metrics typically rely on similarity measures. Therefore, experts have not reached a consensus over a general similarity measure that would be appropriate for a wide range of problems.

In response to this challenge, instead of developing a model that explicitly satisfies a notion of fairness, I propose an indirect method that implicitly guarantees fairness. The proposed process, which I refer to it as "democratization of fairness in machine learning," solicits judgments on the fairness of the model's decisions from users affected by these decisions. Maximizing the positive feedbacks of users can indirectly result in the fairness of the model decision. Currently, fairness in ML suffers from a lack of direct conversation with users (subjects) affected by the model's decisions. The democratization of fairness in ML works to close this gap through two main steps. The first step is to ensure that subjects understand the algorithm decision-making process. As a result of this understanding, well-informed subjects can have more reliable judgments about algorithm decisions' fairness. The second step is to update the model based on the feedback of the well-informed subjects.

Step one in the democratization of fairness is related to the explainability topic. Fairness and explainability concepts are intertwined, and explanations, which try to help human users understand the functionality of the ML models, can be utilized to reach fairness in ML models [12]. Previous research has shown that the explanation of an algorithmic decision process can help detect discrimination behavior[20]. For instance, counterfactual explanations ask questions like, "Had an individual been of a different protected status, would the model have treated them differently?"[20] and then use the answer as a fairness measure. However, the kind of explanation depends on who is the target of the explanation. In the future, I aim to understand how to utilize an explanation as a tool for involving the subjects in the process of detecting discrimination.

Step two in the democratization of fairness is integrating the subjects' feedback into the training and updating of the model. There is a line of research in fairness in recent years using auditing for detecting discrimination against subgroups of the population. In this body of research, auditors are either an algorithm playing a zero-sum game against the model [86] or they are a group of experts who have the required skillset to make a judgment [150]. However, none of this research considers placing the users affected by the model decisions in the position of Auditor. I am interested in expanding the idea of auditing to include these users.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Developer agreement and policy – twitter developers | twitter developer platform.

[2] Goldman sachs' misguided world cup predictions could provide clues to the apple card controversy.

[3] Abello, J., Resende, M. G., and Sudarsky, S. Massive quasi-clique detection. In *Latin American symposium on theoretical informatics* (2002), Springer, pp. 598–612.

[4] Adams, J. S. Inequity in social exchange. In *Advances in experimental social psychology*, vol. 2. Elsevier, 1965, pp. 267–299.

[5] Agarwal, C., Lakkaraju*, H., and Zitnik*, M. Towards a unified framework for fair and stable graph representation learning.

[6] Agrawal, A., Verschueren, R., Diamond, S., and Boyd, S. A rewriting system for convex optimization problems. *Journal of Control and Decision 5*, 1 (2018), 42–60.

[7] Akoglu, L., McGlohon, M., and Faloutsos, C. Oddball: Spotting anomalies in weighted graphs. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (2010), Springer, pp. 410–421.

[8] Akoglu, L., Tong, H., and Koutra, D. Graph based anomaly detection and description: a survey. *Data mining and knowledge discovery 29*, 3 (2015), 626–688.

[9] Al Hasan, M., Chaoji, V., Salem, S., and Zaki, M. Link prediction using supervised learning. In *SDM06: workshop on link analysis, counter-terrorism and security* (2006).

[10] Ali, M., Sapiezynski, P., Bogen, M., Korolova, A., Mislove, A., and Rieke, A. Discrimination through optimization: How facebook's ad delivery can lead to biased outcomes. *Proceedings of the ACM on Human-Computer Interaction 3*, CSCW (2019), 1–30.

[11] Angwin, J., Larson, J., Mattu, S., and Kirchner, L. Machine bias. *ProPublica, May 23* (2016).

[12] Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion 58* (2020), 82–115.

[13] Bayati, M., Gerritsen, M., Gleich, D. F., Saberi, A., and Wang, Y. Algorithms for large, sparse network alignment problems. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on* (2009), IEEE, pp. 705–710.

[14] Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence 35*, 8 (2013), 1798–1828.

[15] Berk, R. *Criminal justice forecasts of risk: A machine learning approach*. Springer Science & Business Media, 2012.

[16] Berk, R., Heidari, H., Jabbari, S., Kearns, M., and Roth, A. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research* (2018).

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

[18] Bhardwaj, R., Nambiar, A. R., and Dutta, D. A study of machine learning in healthcare. In *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)* (2017), vol. 2, IEEE, pp. 236–241.

[19] Bian, A. A., Buhmann, J. M., Krause, A., and Tschiatschek, S. Guarantees for greedy maximization of non-submodular functions with applications. In *International conference on machine learning* (2017), PMLR, pp. 498–507.

[20] Black, E., Yeom, S., and Fredrikson, M. Fliptest: fairness testing via optimal transport. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (2020), pp. 111–121.

[21] Boldi, P., Luongo, A., and Vigna, S. Rank monotonicity in centrality measures.

[22] Boldi, P., and Vigna, S. Axioms for centrality. *Internet Mathematics 10*, 3-4 (2014), 222–262.

[23] Buyl, M., and De Bie, T. Debayes: a bayesian method for debiasing network embeddings. In *International Conference on Machine Learning* (2020), PMLR, pp. 1220–1229.

[24] Calders, T., Kamiran, F., and Pechenizkiy, M. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops* (2009).

[25] Chang, J., and Blei, D. Relational topic models for document networks. In *Artificial Intelligence and Statistics* (2009), pp. 81–88.

[26] Charlin, L., and Zemel, R. The toronto paper matching system: an automated paper-reviewer assignment system.

[27] Chen, J., Geyer, W., Dugan, C., Muller, M., and Guy, I. Make new friends, but keep the old: recommending people on social networking sites. In *SIGCHI* (2009).

[28] Chen, J., Ma, T., and Xiao, C. Fastgcn: fast learning with graph convolutional networks via importance sampling. *arXiv preprint arXiv:1801.10247* (2018).

[29] Chen, L., Feldman, M., and Karbasi, A. Weakly submodular maximization beyond cardinality constraints: Does randomization help greedy? In *International Conference on Machine Learning* (2018), PMLR, pp. 804–813.

[30] Chen, R., Hua, Q., Chang, Y.-S., Wang, B., Zhang, L., and Kong, X. A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks. *IEEE Access 6* (2018), 64301–64320.

[31] Chiang, W.-L., Liu, X., Si, S., Li, Y., Bengio, S., and Hsieh, C.-J. Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2019), pp. 257–266.

[32] Choi, E., Bahadori, M. T., Song, L., Stewart, W. F., and Sun, J. Gram: graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2017), pp. 787–795.

[33] Chouldechova, A. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data 5*, 2 (2017), 153–163.

[34] Clauset, A., Moore, C., and Newman, M. E. Hierarchical structure and the prediction of missing links in networks. *arXiv preprint arXiv:0811.0484* (2008).

[35] Cong, W., Forsati, R., Kandemir, M., and Mahdavi, M. Minimal variance sampling with provable guarantees for fast training of graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2020), pp. 1393–1403.

[36] Corbett-Davies, S., and Goel, S. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023* (2018).

[37] Corbett-Davies, S., Pierson, E., Feller, A., Goel, S., and Huq, A. Algorithmic decision making and the cost of fairness. In *SIGKDD* (2017), ACM, pp. 797–806.

[38] Cramér, H., and Leadbetter, M. R. *Stationary and related stochastic processes: Sample function properties and their applications*. Courier Corporation, 2013.

[39] Crandall, D. J., Backstrom, L., Cosley, D., Suri, S., Huttenlocher, D., and Kleinberg, J. Inferring social ties from geographic coincidences. *PNAS 107*, 52 (2010), 22436–22441.

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

[41] Dai, Q., Li, Q., Tang, J., and Wang, D. Adversarial network embedding. In *Thirty-Second AAAI Conference on Artificial Intelligence* (2018).

[42] Dastin, J. Amazon scraps secret ai recruiting tool that showed bias against women, Oct 2018.

[43] Davis, G., Mallat, S., and Avellaneda, M. Adaptive greedy approximations. *Constructive approximation 13*, 1 (1997), 57–98.

[44] De Sá, H. R., and Prudêncio, R. B. Supervised link prediction in weighted networks. In *The 2011 international joint conference on neural networks* (2011), IEEE, pp. 2281–2288.

[45] Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. Convolutional 2d knowledge graph embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence* (2018).

[46] Diamond, S., and Boyd, S. Cvxpy: A python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research 17*, 1 (2016), 2909–2913.

[47] Dieterich, W., Mendoza, C., and Brennan, T. Compas risk scales: Demonstrating accuracy equity and predictive parity. *Northpoint Inc 7*, 7.4 (2016), 1.

[48] Dua, C. Dheeru and graff,“. *UCI Machine Learning Repository,” UCI Machine Learning Repository* (2017).

[49] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference* (2012).

[50] Edwards, H., and Storkey, A. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897* (2015).

[51] Eldardiry, H., and Neville, J. A resampling technique for relational data graphs. In *Proceedings of the 2nd SNA workshop, 14th ACM SIGKDD conference on knowledge discovery and data mining* (2008).

[52] Fakhraei, S., Foulds, J., Shashanka, M., and Getoor, L. Collective spammer detection in evolving multi-relational social networks. In *SIGKDD* (2015), ACM, pp. 1769–1778.

[53] Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. Certifying and removing disparate impact. In *SIGKDD* (2015), ACM, pp. 259–268.

[54] Feng, J., Huang, M., Yang, Y., and Zhu, X. Gake: Graph aware knowledge embedding. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (2016), pp. 641–651.

[55] Fey, M., Eric Lenssen, J., Weichert, F., and Müller, H. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018), pp. 869–877.

[56] Fiez, T., Shah, N., and Ratliff, L. A super* algorithm to optimize paper bidding in peer review. In *Conference on Uncertainty in Artificial Intelligence* (2020), PMLR, pp. 580–589.

[57] Frank, O., and Snijders, T. Estimating the size of hidden populations using snowball sampling. *JOURNAL OF OFFICIAL STATISTICS-STOCKHOLM- 10* (1994), 53–53.

[58] Friedler, S. A., Scheidegger, C., and Venkatasubramanian, S. On the (im) possibility of fairness. *arXiv preprint arXiv:1609.07236* (2016).

[59] Fu, X., Ren, X., Mengshoel, O. J., and Wu, X. Stochastic optimization for market return prediction using financial knowledge graph. In *2018 IEEE International Conference on Big Knowledge (ICBK)* (2018), IEEE, pp. 25–32.

[60] Gilbert, A. C., Muthukrishnan, S., and Strauss, M. J. Approximation of functions over redundant dictionaries using coherence. In *SODA* (2003), Citeseer, pp. 243–252.

[61] Gjoka, M., Kurant, M., Butts, C. T., and Markopoulou, A. Walking in facebook: A case study of unbiased sampling of osns. In *2010 Proceedings IEEE Infocom* (2010), Ieee, pp. 1–9.

[62] Goel, S., Rao, J. M., and Shroff, R. Personalized risk assessments in the criminal justice system. *American Economic Review 106*, 5 (2016), 119–23.

[63] Goldberg, A. V., and Tarjan, R. E. A new approach to the maximum-flow problem. *Journal of the ACM (JACM) 35*, 4 (1988), 921–940.

[64] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *NeurIPS* (2014), pp. 2672–2680.

[65] Grover, A., and Leskovec, J. node2vec: Scalable feature learning for networks. In *Proc of KDD* (2016), pp. 855–864.

[66] Gupte, M., and Eliassi-Rad, T. Measuring tie strength in implicit social networks. In *Proceedings of the 4th Annual ACM Web Science Conference* (2012), pp. 109–118.

[67] Hamilton, W. L., Ying, R., and Leskovec, J. Inductive representation learning on large graphs. In *31st Conf. on Neural Information Processing Systems (NIPS 2017)* (2017).

[68] Han, X., Liu, Z., and Sun, M. Neural knowledge acquisition via mutual attention between knowledge graph and text. In *Thirty-Second AAAI Conference on Artificial Intelligence* (2018).

[69] Hardt, M., Price, E., Srebro, N., et al. Equality of opportunity in supervised learning. In *NeurIPS* (2016), pp. 3315–3323.

[70] Hébert-Johnson, U., Kim, M. P., Reingold, O., and Rothblum, G. N. Calibration for the (computationally-identifiable) masses. *arXiv preprint arXiv:1711.08513* (2017).

[71] Henderson, K., Eliassi-Rad, T., Faloutsos, C., Akoglu, L., Li, L., Maruhashi, K., Prakash, B. A., and Tong, H. Metric forensics: a multi-level approach for mining volatile graphs. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (2010), pp. 163–172.

[72] Hofstra, B., Corten, R., Van Tubergen, F., and Ellison, N. B. Sources of segregation in social networks: A novel approach using facebook. *American Sociological Review 82*, 3 (2017), 625–656.

[73] Hu, P., and Lau, W. C. A survey and taxonomy of graph sampling. *arXiv preprint arXiv:1308.5865* (2013).

[74] Huang, X., Li, J., and Hu, X. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* (2017), ACM, pp. 731–739.

[75] Hübler, C., Kriegel, H.-P., Borgwardt, K., and Ghahramani, Z. Metropolis algorithms for representative subgraph sampling. In *2008 Eighth IEEE International Conference on Data Mining* (2008), IEEE, pp. 283–292.

[76] Jecmen, S., Zhang, H., Liu, R., Shah, N. B., Conitzer, V., and Fang, F. Mitigating manipulation in peer review via randomized reviewer assignments. *arXiv preprint arXiv:2006.16437* (2020).

[77] Jefferson, B. J. Predictable policing: Predictive crime mapping and geographies of policing and race. *Annals of the American Association of Geographers 108*, 1 (2018), 1–16.

[78] Johndrow, J. E., Lum, K., et al. An algorithm for removing sensitive information: application to race-independent recidivism prediction. *The Annals of Applied Statistics 13*, 1 (2019), 189–220.

[79] Kamiran, F., Karim, A., and Zhang, X. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining* (2012), IEEE, pp. 924–929.

[80] Kamishima, T., Akaho, S., and Sakuma, J. Fairness-aware learning through regularization approach. In *ICDM Workshops* (2011), IEEE, pp. 643–650.

[81] Kang, J., He, J., Maciejewski, R., and Tong, H. Inform: Individual fairness on graph mining. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2020), pp. 379–389.

[82] Kang, U., Tsourakakis, C. E., Appel, A. P., Faloutsos, C., and Leskovec, J. Hadi: Mining radii of large graphs. *ACM Transactions on Knowledge Discovery from Data (TKDD) 5*, 2 (2011), 1–24.

[83] Karimi, F., Génois, M., Wagner, C., Singer, P., and Strohmaier, M. Homophily influences ranking of minorities in social networks. *Scientific reports 8* (2018).

[84] Kaur, R., and Singh, S. A survey of data mining and social network analysis based anomaly detection techniques. *Egyptian informatics journal 17*, 2 (2016), 199–216.

[85] Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. *arXiv preprint arXiv:1711.05144* (2017).

[86] Kearns, M., Neel, S., Roth, A., and Wu, Z. S. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International Conference on Machine Learning* (2018), PMLR, pp. 2564–2572.

[87] Kelly, J., Sadeghieh, T., and Adeli, K. Peer review in scientific publications: benefits, critiques, & a survival guide. *EJIFCC 25*, 3 (2014), 227.

[88] Kingma, D., and Ba, J. Adam: A method for stochastic optimization. In *ICLR* (2014).

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

[90] Kipf, T. N., and Welling, M. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308* (2016).

[91] Klau, G. W. A new graph-based method for pairwise global network alignment. *BMC bioinformatics 10*, 1 (2009), S59.

[92] Kleinberg, J., Mullainathan, S., and Raghavan, M. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807* (2016).

[93] Kobren, A., Saha, B., and McCallum, A. Paper matching with local fairness constraints. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2019), pp. 1247–1257.

[94] Kollias, G., Mohammadi, S., and Grama, A. Network similarity decomposition (nsd): A fast and scalable approach to network alignment. *IEEE Transactions on Knowledge and Data Engineering 24*, 12 (2011), 2232–2243.

[95] Koutra, D., Tong, H., and Lubensky, D. Big-align: Fast bipartite graph alignment. In *2013 IEEE 13th International Conference on Data Mining* (2013), IEEE, pp. 389–398.

[96] Kuchaiev, O., Milenković, T., Memišević, V., Hayes, W., and Pržulj, N. Topological network alignment uncovers biological function and phylogeny. *Journal of the Royal Society Interface* (2010), rsif20100063.

[97] Kusner, M. J., Loftus, J., Russell, C., and Silva, R. Counterfactual fairness. In *NeurIPS* (2017), pp. 4066–4076.

[98] Lakshmi, T. J., and Bhavani, S. D. Link prediction in temporal heterogeneous networks. In *Pacific-Asia Workshop on Intelligence and Security Informatics* (2017), Springer, pp. 83–98.

[99] Lambrecht, A., and Tucker, C. Algorithmic bias? an empirical study of apparent gender-based discrimination in the display of stem career ads. *Management Science 65*, 7 (2019), 2966–2981.

[100] Laniado, D., Volkovich, Y., Kappler, K., and Kaltenbrunner, A. Gender homophily in online dyadic and triadic relationships. *EPJ Data Science 5*, 1 (2016), 19.

[101] Lashbrook, A. Ai-driven dermatology could leave dark-skinned patients behind, Aug 2018.

[102] Lee, M. K., and Baykal, S. Algorithmic mediation in group decisions: Fairness perceptions of algorithmically mediated vs. discussion-based social division. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (2017), pp. 1035–1048.

[103] Leo, M., Sharma, S., and Maddulety, K. Machine learning in banking risk management: A literature review. *Risks 7*, 1 (2019), 29.

[104] Leskovec, J., and Faloutsos, C. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (2006), pp. 631–636.

[105] Leskovec, J., Kleinberg, J., and Faloutsos, C. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining* (2005), pp. 177–187.

[106] Leskovec, J., and Mcauley, J. J. Learning to discover social circles in ego networks. In *NeurIPS* (2012), pp. 539–547.

[107] Li, K., Gao, J., Guo, S., Du, N., Li, X., and Zhang, A. Lrbm: A restricted boltzmann machine based approach for representation learning on linked data. In *Data Mining (ICDM), 2014 IEEE International Conference on* (2014), IEEE, pp. 300–309.

[108] Li, X., Du, N., Li, H., Li, K., Gao, J., and Zhang, A. A deep learning approach to link prediction in dynamic networks. In *Proceedings of the 2014 SIAM International Conference on Data Mining* (2014), SIAM, pp. 289–297.

[109] Li, Y., Tarlow, D., Brockschmidt, M., and Zemel, R. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* (2015).

[110] Liao, C.-S., Lu, K., Baym, M., Singh, R., and Berger, B. Isorankn: spectral methods for global alignment of multiple protein networks. *Bioinformatics 25*, 12 (2009), i253–i258.

[111] Liben-Nowell, D., and Kleinberg, J. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology 58*, 7 (2007), 1019–1031.

[112] Lichtenwalter, R. N., Lussier, J. T., and Chawla, N. V. New perspectives and methods in link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining* (2010), pp. 243–252.

[113] Louizos, C., Swersky, K., Li, Y., Welling, M., and Zemel, R. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830* (2015).

[114] Lu, J., Wu, D., Mao, M., Wang, W., and Zhang, G. Recommender system application developments: a survey. *Decision Support Systems 74* (2015), 12–32.

[115] Lu, Q., and Getoor, L. Link-based classification. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)* (2003), pp. 496–503.

[116] Madras, D., Creager, E., Pitassi, T., and Zemel, R. Learning adversarially fair and transferable representations. *arXiv preprint arXiv:1802.06309* (2018).

[117] Mariani, M. S., Medo, M., and Zhang, Y.-C. Ranking nodes in growing networks: When pagerank fails. *Scientific reports 5*, 1 (2015), 1–10.

[118] Martinez, N., Bertran, M., and Sapiro, G. Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning* (2020), PMLR, pp. 6755–6764.

[119] Masrour, F., Barjesteh, I., Forsati, R., Esfahanian, A.-H., and Radha, H. Network completion with node similarity: A matrix completion approach with provable guarantees. In *Advances in Social Networks Analysis and Mining (ASONAM), 2015 IEEE/ACM International Conference on* (2015), IEEE, pp. 302–307.

[120] Masrour, F., Tan, P.-N., Esfahanian, A.-H., and VanDam, C. Attributed network representation learning approaches for link prediction. In *ASONAM* (2018), IEEE, pp. 560–563.

[121] Masrour, F., Wilson, T., Yan, H., Tan, P.-N., and Esfahanian, A. Bursting the filter bubble: Fairness-aware network link prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2020), vol. 34, pp. 841–848.

[122] Massoulié, L., Le Merrer, E., Kermarrec, A.-M., and Ganesh, A. Peer counting and sampling in overlay networks: random walk methods. In *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing* (2006), pp. 123–132.

[123] McPherson, M., Smith-Lovin, L., and Cook, J. M. Birds of a feather: Homophily in social networks. *Annual review of sociology 27*, 1 (2001), 415–444.

[124] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., and Galstyan, A. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635* (2019).

[125] Menon, A., and Elkan, C. Link prediction via matrix factorization. *Machine Learning and Knowledge Discovery in Databases* (2011), 437–452.

[126] Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[127] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (2013), pp. 3111–3119.

[128] Mitzenmacher, M. A brief history of lognormal and power law distributions. In *Proceedings of the Allerton conference on communication, control, and computing* (2001), pp. 182–191.

[129] Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming 14*, 1 (1978), 265–294.

[130] Newman, M. *Networks: an introduction.* Oxford university press, 2010.

[131] Newman, M. E. Clustering and preferential attachment in growing networks. *Physical review E 64*, 2 (2001), 025102.

[132] Newman, M. E. Finding community structure in networks using the eigenvectors of matrices. *Physical review E 74*, 3 (2006), 036104.

[133] Newman, M. E. Modularity and community structure in networks. *PNAS 103*, 23 (2006), 8577–8582.

[134] Newman, M. E., and Girvan, M. Finding and evaluating community structure in networks. *Physical review E 69*, 2 (2004), 026113.

[135] Nguyen, T. T., Hui, P.-M., Harper, F. M., Terveen, L., and Konstan, J. A. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *WWW* (2014).

[136] O'Neil, C. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books, 2017.

[137] Özcan, A., and Öğüdücü, Ş. G. Multivariate temporal link prediction in evolving social networks. In *2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS)* (2015), IEEE, pp. 185–190.

[138] Page, L. Method for node ranking in a linked database, Sept. 4 2001. US Patent 6,285,999.

[139] Page, L., Brin, S., Motwani, R., and Winograd, T. The pagerank citation ranking: Bringing order to the web. Tech. rep., Stanford InfoLab, 1999.

[140] Pan, S., Hu, R., Long, G., Jiang, J., Yao, L., and Zhang, C. Adversarially regularized graph autoencoder for graph embedding. *arXiv preprint arXiv:1802.04407* (2018).

[141] Pariser, E. The filter bubble: What the internet is hiding, 2012.

[142] Pattillo, J., Youssef, N., and Butenko, S. Clique relaxation models in social network analysis. In *Handbook of Optimization in Complex Networks*. Springer, 2012, pp. 143–162.

[143] Peiró, J. M., Martínez-Tur, V., and Moliner, C. *Perceived Fairness*. Springer Netherlands, Dordrecht, 2014, pp. 4693–4696.

[144] Perozzi, B., Al-Rfou, R., and Skiena, S. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (2014), pp. 701–710.

[145] Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. On fairness and calibration. In *NeurIPS* (2017).

[146] Qian, C., Li, G., Feng, C., and Tang, K. Distributed pareto optimization for subset selection. In *IJCAI* (2018), pp. 1492–1498.

[147] Rahman, T. A., Surma, B., Backes, M., and Zhang, Y. Fairwalk: Towards fair graph embedding. In *IJCAI* (2019), pp. 3289–3295.

[148] Ribeiro, B., and Towsley, D. Estimating and sampling graphs with multidimensional random walks. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (2010), pp. 390–403.

[149] Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S., and Sontag, D. Learning a health knowledge graph from electronic medical records. *Scientific reports 7*, 1 (2017), 1–11.

[150] Saleiro, P., Kuester, B., Hinkson, L., London, J., Stevens, A., Anisfeld, A., Rodolfa, K. T., and Ghani, R. Aequitas: A bias and fairness audit toolkit. *arXiv preprint arXiv:1811.05577* (2018).

[151] Saxena, N., Huang, K., DeFilippis, E., Radanovic, G., Parkes, D., and Liu, Y. How do fairness definitions fare? examining public attitudes towards algorithmic definitions of fairness, 2018.

[152] Scripps, J., Tan, P.-N., Chen, F., and Esfahanian, A.-H. A matrix alignment approach for link prediction. In *Proc of ICPR* (2008).

[153] Shulga, L., and Tanford, S. Measuring perceptions of fairness of loyalty program members. *Journal of Hospitality Marketing & Management 27*, 3 (2018), 346–365.

[154] Singh, R., Xu, J., and Berger, B. Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proceedings of the National Academy of Sciences 105*, 35 (2008), 12763–12768.

[155] Snijders, T. A., Van de Bunt, G. G., and Steglich, C. E. Introduction to stochastic actor-based models for network dynamics. *Social networks 32*, 1 (2010), 44–60.

[156] Stelmakh, I., Shah, N., and Singh, A. On testing for biases in peer review. In *Advances in Neural Information Processing Systems* (2019), pp. 5286–5296.

[157] Stelmakh, I., Shah, N. B., and Singh, A. Peerreview4all: Fair and accurate reviewer assignment in peer review. *arXiv preprint arXiv:1806.06237* (2018).

[158] Stelmakh, I., Shah, N. B., and Singh, A. Peerreview4all: Fair and accurate reviewer assignment in peer review. In *Algorithmic Learning Theory* (2019), PMLR, pp. 828–856.

[159] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., and Mei, Q. Line: Large-scale information network embedding. In *Proc of WWW* (2015), pp. 1067–1077.

[160] Tian, F., Gao, B., Cui, Q., Chen, E., and Liu, T.-Y. Learning deep representations for graph clustering. In *AAAI* (2014), pp. 1293–1299.

[161] Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological) 58*, 1 (1996), 267–288.

[162] Tomkins, A., Zhang, M., and Heavlin, W. D. Reviewer bias in single-versus double-blind peer review. *Proceedings of the National Academy of Sciences 114*, 48 (2017), 12708–12713.

[163] Turner, A. How many people have smartphones worldwide (sept 2021).

[164] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., and Bengio, Y. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[165] Vogelstein, J. T., Conroy, J. M., Podrazik, L. J., Kratzer, S. G., Fishkind, D. E., Vogelstein, R. J., and Priebe, C. E. Fast inexact graph matching with applications in statistical connectomics. *CoRR, abs/1112.5507* (2011).

[166] Wagner, C., Singer, P., Karimi, F., Pfeffer, J., and Strohmaier, M. Sampling from social networks with attributes. In *Proceedings of the 26th International Conference on World Wide Web* (2017), pp. 1181–1190.

[167] Wang, D., Pedreschi, D., Song, C., Giannotti, F., and Barabasi, A.-L. Human mobility, social ties, and link prediction. In *SIGKDD* (2011), Acm, pp. 1100–1108.

[168] Wang, F., and Zhang, C. Label propagation through linear neighborhoods. *IEEE Transactions on Knowledge and Data Engineering 20*, 1 (2007), 55–67.

[169] Wang, H., Wang, J., Wang, J., Zhao, M., Zhang, W., Zhang, F., Xie, X., and Guo, M. Graphgan: Graph representation learning with generative adversarial nets. In *Thirty-second AAAI conference on artificial intelligence* (2018).

[170] Wang, J., and Shah, N. B. Your 2 is my 1, your 3 is my 9: Handling arbitrary miscalibrations in ratings. *arXiv preprint arXiv:1806.05085* (2018).

[171] Wang, P., Xu, B., Wu, Y., and Zhou, X. Link prediction in social networks: the state-of-the-art. *Science China Information Sciences 58*, 1 (2015), 1–38.

[172] Wang, R., Li, B., Hu, S., Du, W., and Zhang, M. Knowledge graph embedding via graph attenuated attention networks. *IEEE Access* (2019).

[173] Washington, A. L. How to argue with an algorithm: Lessons from the compas-propublica debate. *Colo. Tech. LJ 17* (2018), 131.

[174] Wei, W., Erenrich, J., and Selman, B. Towards efficient sampling: Exploiting random walk strategies. In *AAAI* (2004), vol. 4, pp. 670–676.

[175] Wiens, J., and Shenoy, E. S. Machine learning for healthcare: on the verge of a major shift in healthcare epidemiology. *Clinical Infectious Diseases 66*, 1 (2018), 149–153.

[176] Xie, T., Ma, Y., Tong, H., Thai, M. T., and Maciejewski, R. Auditing the sensitivity of graph-based ranking with visual analytics. *IEEE Transactions on Visualization and Computer Graphics* (2020).

[177] Xiong, C., Power, R., and Callan, J. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th international conference on world wide web* (2017), pp. 1271–1279.

[178] Yang, C., Liu, Z., Zhao, D., Sun, M., and Chang, E. Y. Network representation learning with rich text information. In *IJCAI* (2015), pp. 2111–2117.

[179] Yang, J., and Leskovec, J. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining* (2013), pp. 587–596.

[180] Yang, J., McAuley, J., and Leskovec, J. Community detection in networks with node attributes. In *2013 IEEE 13th International Conference on Data Mining* (2013), IEEE, pp. 1151–1156.

[181] Yang, T., Jin, R., Chi, Y., and Zhu, S. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (2009), pp. 927–936.

[182] Yeh, I.-C., and Lien, C.-h. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications 36*, 2 (2009), 2473–2480.

[183] Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W. L., and Leskovec, J. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2018), pp. 974–983.

[184] Zafarani, R., Abbasi, M. A., and Liu, H. *Social media mining: an introduction*. Cambridge University Press, 2014.

[185] Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., and Baeza-Yates, R. Fa* ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (2017), pp. 1569–1578.

[186] Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. Learning fair representations. In *ICML* (2013), pp. 325–333.

[187] Zeng, H., Zhou, H., Srivastava, A., Kannan, R., and Prasanna, V. Graphsaint: Graph sampling based inductive learning method. *arXiv preprint arXiv:1907.04931* (2019).

[188] Zhai, S., and Zhang, Z. Dropout training of matrix factorization and autoencoder for link prediction in sparse graphs. In *Proceedings of the 2015 SIAM International Conference on Data Mining* (2015), SIAM, pp. 451–459.

[189] Zhang, S., and Tong, H. Final: Fast attributed network alignment. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), ACM, pp. 1345–1354.

[190] Zhang, X., Liu, H., Li, Q., and Wu, X.-M. Attributed graph clustering via adaptive graph convolution. *arXiv preprint arXiv:1906.01210* (2019).

[191] Zhang, Y., Yao, Q., Dai, W., and Chen, L. Autokge: searching scoring functions for knowledge graph embedding. *arXiv preprint arXiv:1904.11682* (2019).

[192] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. Learning with local and global consistency. In *Advances in neural information processing systems* (2004), pp. 321–328.

[193] Zhu, X., Ghahramani, Z., and Lafferty, J. D. Semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the 20th International conference on Machine learning (ICML-03)* (2003), pp. 912–919.

[194] Zou, D., Hu, Z., Wang, Y., Jiang, S., Sun, Y., and Gu, Q. Layer-dependent importance sampling for training deep and large graph convolutional networks. In *Advances in Neural Information Processing Systems* (2019), pp. 11249–11259.

[195] Zou, H., and Hastie, T. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology) 67*, 2 (2005), 301–320.