Applying Fairness Constraints on Graph Node Ranks under Personalization Bias

Emmanouil Krasanakis, Symeon Papadopoulos, and Ioannis Kompatsiaris

CERTH-ITI, 57001, Thessaloniki, Greece
{maniospas,papadop,ikom}@iti.gr

Abstract. In this work we address algorithmic fairness concerns that arise when graph nodes are ranked based on their structural relatedness to a personalized set of query ones. In particular, we aim to mitigate disparate impact, i.e. the difference in average rank between nodes of a sensitive attribute compared to the rest, while also preserving node rank quality. To do this, we introduce a personalization editing mechanism whose parameters can be adjusted to help the ranking algorithm achieve a variety of trade-offs between fairness constraints and rank changes. In experiments across three real-world social graphs and two base ranking algorithms, our approach outperforms baseline and existing methods in uniformly mitigating disparate impact, even when personalization suffers from extreme bias. In particular, it achieves higher trade-offs between fairness and rank quality and manages to preserve most of node rank quality when a constrained amount of disparate impact is allowed.

Keywords: node ranking, personalized ranking, algorithmic fairness, disparate impact mitigation

1 Introduction

Machine learning has been widely adopted in systems that affect important aspects of people's lives, from recommending social media friends to assisting jurisdictional or employment decisions. Since these systems often learn to replicate human-generated and systemic real-world biases, fairness concerns arise when the outcome of automated decisions end up correlating to sensitive attributes, such as gender or ethnicity [1, 2]. Approaches commonly define fairness as similar assessment between sensitive and non-sensitive groups of data entries under a statistical measure [1, 3–5]. In this work, we focus on disparate impact elimination [6–9], which requires (approximate) statistical parity between sensitive and non-sensitive positive predictions.

Node ranking is a type of machine learning that organizes relational data into graphs, whose nodes are ranked based on their structural relatedness to a subset of query ones. Ranking can be *personalized*, in the sense that query nodes share an attribute (e.g. the same political views), in which case node ranks can be used as estimators for that attribute [10–12]. If no personalization takes place and all nodes are used as queries, ranks reflect the structural importance of nodes [13, 14].

Although graph ranking is an important machine learning discipline, remarkably little work has been done to make it fair. In fact, the first -to our knowledge-principled understanding of node rank fairness was only recently proposed by Tsioutsiouliklis et al. [15], who explore disparate impact mitigation for the node ranks of Google's non-personalized PageRank algorithm [16]. We now initiate a discussion on the fairness of personalized node ranking algorithms. Contrary to non-personalized algorithms, where node rank quality is tied to ad hoc definitions of structural importance, in this case there exist objective notions of node rank quality that fairness-aware approaches should ideally respect.

In this work we refer to a convex permutation model for estimating the unfairness of data entries [3] and adapt it to estimate an unbiased personalization that yields similar yet fairer node ranks. This model can be trained towards a variety of fairness-aware objectives, such as fully eliminating disparate impact or minimizing rank edits under statistical parity constraints. We corroborate its efficacy by comparing it to baseline and existing practices across two ranking algorithms and three real-world graphs with both unbiased and extremely biased personalization.

Our contribution lies in initiating a discussion on fairness-aware personalized ranking algorithms, where we address biased personalization and the preservation of prediction-related node rank quality. Furthermore, we investigate whether approaches uniformly introduce fairness in the sense that they do so for both the whole graph and an evaluation subset of nodes.

2 Background

2.1 Personalized Node Ranking Algorithms

Personalized node ranking starts from a set of query nodes sharing an attribute of interest and scores nodes per some notion of structural proximity to query ones. We organize node scores, which are called *ranks*, into vectors r whose elements $r[v] \geq 0$ correspond to nodes v. We similarly organize a personalization vector p, whose elements $p[v] \in [0, 1]$ reflect the probability of nodes v being query ones.

Ranking algorihms are often expressed as graph filters [17, 18]. These use a normalization W of the graph's adjacency matrix, whose elements $W[u, v]$ define transitions from nodes u to v. Then, given that propagating the personalization n hops away can be written as $W^n p$, they weigh different propagation distances:

$$\begin{aligned} r &= H(W)p \\ H(W) &= \sum_{n=0}^{\infty} h_n W^n \end{aligned} \tag{1}$$

$H(W)$ is a graph filter. Different filters can be obtained for different weights h_n methods of calculating W. For example, the graph's adjacency matrix M

can be normalized column-wise $W = MD^{-1}$ or symmetrically $D^{-\frac{1}{2}}MD^{-\frac{1}{2}}$, where $D = diag\big(\big[\sum_u M[v,u]\big]_v\big)$ is the diagonal table of node degrees. Two well-known graph filters are Personalized PageRank [19, 20] and Heat Kernels [21], which respectively arise from hop weights $h_n = (1-a)a^n$ and $h_n = e^{-t}t^n/n!$ for parameters $a \in [0,1]$ and $t \in \{1,2,3,\dots\}$.

The sweep procedure [22, 23] is a method that utilizes node ranking algorithms to identify tightly-knit congregations of nodes well-separated from the rest of the graph, a concept known as subgraph conductance [24]. This method assumes that a base ranking algorithm R with strong locality [25], such as personalized PageRank and Heat Kernels, yields ranks $R(p)$ for a personalization p that comprises structurally close query nodes. It then compares ranks with their non-personalized counterparts $R(\mathbf{1})$, where $\mathbf{1}$ is a vector of ones:

$$r_{sweep} = \frac{R(p)[v]}{R(\mathbf{1})[v]} \tag{2}$$

From now on, we will refer to this postprocessing as the *sweep ratio*.

The sweep procedure orders all nodes based on their sweep ratio and cuts the graph into two partitions so that conductance is minimized. This practice statistically yields well-separated partitions for a variety of node ranking algorithms [22–24]. From a high-level perspective, this indicates that the sweep ratio tends to improve node rank quality.

2.2 Algorithmic Fairness and Graph Mining

Algorithmic fairness is broadly understood as parity between sensitive and non-sensitive group entries, in the sense that a chosen statistical property is not biased in favor of either. Three popular fairness-aware objectives commonly recognized in the literature [4, 1, 3, 5] are disparate treatment elimination, disparate impact elimination and disparate mistreatment elimination. These correspond to not using the sensitive attribute in predictions, preserving statistical parity between the fraction of sensitive and non-sensitive positive labels and achieving identical predictive performance on the two groups under a measure of choice.

In this work, we focus on disparate impact [6–9, 1] as the type of unfairness to mitigate. A well-established measure that quantifies how well a system mitigates disparate impact is the *pRule* [6]; denoting as $R[v]$ the binary outputs of a system R for entries v as $R[v]$, S the sensitive group and $P(a|b)$ the probability of a conditioned on b, this measure is defined through as:

$$\begin{aligned} pRule &= \frac{\min(p_S, p_{S'})}{\max(p_S, p_{S'})} \in [0,1] \\ p_S &= P(R[v]=1|p \in S) \\ p_{S'} &= P(R[v]=1|p \notin S) \end{aligned} \tag{3}$$

The higher the pRule, the fairer a system is. There is precedence [6] for considering 80% pRule or higher as fair. Calders-Verwer disparity $|p_S - p_{S'}|$ [7] is a correlated but less descriptive measure that is optimized at the same point.

In domains related to ranking, fairness has been defined for the order of data entry recommendations [26–29] as equity in the ranking positions between sensitive and non-sensitive entries. However, these notions of fairness are not applicable to the more granular understanding provided by node ranks.

In graphs, the notion of achieving fair node embedding has been proposed [30, 31]. These are the first approaches that introduce fair random walks, which are stochastic process modeled by personalized PageRank, although the fairness of these walks is only implicitly asserted through embedding fairness. A more advanced understanding has been achieved recently in the more general domain of graph neural networks [32], which can be trained to produce fair recommendation, even under partial knowledge of the sensitive attribute.

Lastly, a recent work by Tsioutsiouliklis et al. [15] has jump-started a discourse on node rank fairness. Although focused non-personalized ranking, it first recognizes the need of optimizing a trade-off between fairness and preserving rank quality. Furthermore, a first definition of node rank fairness is provided, called *phi*-fairness. Under a stochastic interpretation of node ranks, where they are proportional to the probability of nodes assuming positive labels, ϕ-fairness becomes equivalent to disparate impact elimination when $\phi = \frac{|S|}{|S|+|S'|}$.

In this work we consider the similar objectives of a) trading-off deviation from the original ranks and high pRule and b) preserving rank quality under fairness constraints. The pRule is calculated according to the above-mentioned stochastic interpretation of ranks as:

$$\begin{aligned} p_S &= P(R[v]=1|p\in S) = \frac{\sum_{v\in S} L_\infty(r)[v]}{|S|} \\ p_S &= P(R[v]=1|p\in S) = \frac{\sum_{v\notin S} L_\infty(r)[v]}{|S'|} \end{aligned} \tag{4}$$

where $L_\infty(r)$ is a normalization that divides ranks with their maximum value and R is a stochastic process with probability $P(R[v]=1) = \frac{r[v]}{\max_u r[u]} = L_\infty(r)[v]$.

3 Our Approach

We theorize that there exist two types of potential node rank bias: stationary and rank-related. The first arises when ranks are underestimated or underestimated by the same multiplicative amount. Whereas the depends on the personalization, which transfers either its own or graph edge bias to the ranks. Of the two, stationary bias is easier to treat, as it does not depend on the personalization and only attacks the outcome of the ranking algorithm. In fact, the sweep ratio eliminates it, as it ends up dividing node ranks with their bias term.

On the other hand, rank-related bias is harder to tackle. To see why, let us consider an invertible graph filter, such as the closed form of personalized PageRank $H(W) = (1-a)(I-aW)^{-1}$. For a personalization vector p produces ranks $r = H(W)p$ and there exist fair ranks r_{fair} that satisfy a fairness-aware objective, such as achieving a trade-off between preserving the original ranks and improving the pRule:

$$\text{minimize } \frac{\|L_\infty(r_{fair}) - L_\infty(r)\|_1}{|V|} - pRule(r_{fair})$$

where $pRule(r_{fair})$ calculates the pRule of those ranks across all graph nodes V and $\|\cdot\|_1$ is the L_1 norm that sums the absolute value of vector elements.

In this setting, the personalization $p_{fair} = H^{-1}(W)r_{fair}$ of fair ranks differs from the original one at mast as much as the hard (i.e. achievable) bound:

$$\|p_{fair} - p\| \leq \frac{\|r_{fair} - r\|}{\min_{\lambda \in \text{ eigenvalues of } W} H(\lambda)}$$

where $\|\cdot\|$ is the L_2 vector norm. Depending on the graph filter and adjacency matrix normalization, small deviations between biased node ranks and their fair counterparts can require significant personalization changes to replicate.

Based on the above, searching for a personalization that induces fair ranks is roughly equivalent to directly searching for such ranks, yet computationally harder. Nevertheless, we argue that, if a parametric model with few degrees of freedom is used to edit the personalization, adjusting its parameters to achieve fair ranks would avoid overfitting, i.e. if the edited personalization results in node rank fairness, this would permeate the whole graph in the sense that it would also be achieved by random subsets of nodes.

In this section we propose a model called Fair Personalizer (FP) that can be adjusted to satisfy different fairness-aware node rank objectives. This model's design was motivated by the Convex Underlying Error Permutation (CULEP) we previously developed to reweight training examples [3]. The same practice can not be directly ported to node ranks, since ranking does not inhrently account for negative examples, weighting non-positive personalization vector elements does not change ranking outcome and there exists no validation set for ranks. To address these shorcommings, we move to a dual setting that focuses on correct instead of erroneous node rank identification, shift our focus to editing the personalization and use the personalization as a rough validation set.

We use a stochastic interpretation $\hat{P}(\cdot)$ of ranks that snaps them to 1 with probability proportional to their value and to 0 otherwise. In the rest of this section we avoid adding '$[v]$' next to each quantity and consider all vector operations (including multiplication) to be applied element-by-element.

$$\begin{aligned}
&\hat{P}(r_{fair} = \hat{r}_{fair}) \\
&\quad = \hat{P}(r_{fair} = \hat{r}_{fair} | p = \hat{r}_{fair})\hat{P}(p = \hat{r}_{fair}) \\
&\quad + \hat{P}(r_{fair} = \hat{r}_{fair} | p \neq \hat{r}_{fair})\hat{P}(p \neq \hat{r}_{fair})
\end{aligned}$$

We then borrow CULEP's understanding and estimate the ability of estimated ranks to be fair by perturbing the probability of the estimated personalization being fair. To perform this perturbation, we recognize that, when the former are perfectly estimated, so are the latter. Otherwise, a skewness should be performed that depends on whether and how much ranks overestimate or underestimate the personalization $p[v]$ and whether nodes v are sensitive. Overall, we propose the estimator:

$$\hat{P}(r_{fair} = \hat{r}_{fair}|p = \hat{r}_{fair}) = K\hat{P}(p_{fair} = \hat{p}_{fair})e^{-b(L_\infty(r)-p)}$$
$$\hat{P}(r_{fair} = \hat{r}_{fair}|p \neq \hat{r}_{fair}) = K\hat{P}(p_{fair} = \hat{p}_{fair})e^{b(L_\infty(r)-p)}$$

where b is a vector of real values such that $b[v] = \{b_S \text{ if } v \in S, b_{S'} \text{ otherwise}\}$ and $K \geq 0$ is a common constant. Furthermore, given that selecting sensitive and non-sensitive nodes as part of the personalization is done with fixed probabilities a_S and $a_{S'}$ pertaining to the personalization bias, we organize those into a vector $a = \hat{P}(p = \hat{r}_{fair})$ with elements $a[v] = \{a_S \text{ if } v \in S, a_{S'} \text{ otherwise}\}$. We finally select a fair personalization estimation $\hat{p}_{fair}$ based on a a self-consistency criterion, i.e. that resulting ranks approach fair ones when personalization estimation approaches the respective fairness-inducing personalization:

$$\begin{aligned}\hat{p}_{fair} &= \hat{P}(r_{fair} = \hat{r}_{fair}|p_{fair} = \hat{p}_{fair}) = \frac{\hat{P}(r_{fair} = \hat{r}_{fair})}{\hat{P}(p_{fair} = \hat{p}_{fair})} \\ &\propto ae^{-b(L_\infty(r)-p)} + (1-a)e^{b(L_\infty(r)-p)}\end{aligned} \tag{5}$$

4 Experiment Setup

4.1 Graphs

To assess the ability of our approach to achieve fairness while preserving node rank quality, we experiment on three graphs: two Facebook friendship graphs [33] and one Twitter graph of political retweets [34]. These are chosen on merit that there exists sensitive attribute information for all their nodes.

The Facebook graphs each start from a given user and comprise their social ego network, i.e. the subgraph comprising all social relations between them given and their friends (including relations between friends). Ten such graphs are available in the source material, out of which we randomly select two to experiment on. These are denoted as FacebookX, where X is their starting user. Tens of anonymized binary attributes are available for their nhodes, out of which we consider 'gender' as the sensitive attribute and the first 'education' attribute as the prediction label. The Twitter graph comprises more nodes and edges but only one anonymized sensitive attribute corresponding to each node's binary political opinions (left or right). Due to the lack of a predictive attribute, we define one that as the sensitive attribute's binary complement.

These graphs are overviewed in Table 1. Columns correspond to graph names, number of nodes, number of edges, fraction of nodes with positive labels, number of nodes designated as sensitive and *pRule* value of their positive labels.

Graph	Nodes	Edges	$\frac{Positive}{Nodes}$	$\frac{Sensitive}{Nodes}$	pRule
Facebook0	347	5,038	.68	.36	.91
Facebook686	170	3,312	.55	.46	.91
Twitter	18,470	48,365	.61	.39	0

Table 1. Experiment graph characteristics

4.2 Compared Methods

In our experiments we compare the ability of several methods to bring fairness on personalized PageRank and Heat Kernels. These algorithms were run with the frequently-used parameters $a = .85$ and $t = 3$ and with symmetric normalization, which a preliminary investigation revealed to yield higher values for the AUC measure presented later in this section. Their ranks were computed to a numerical precision of 10^{-9} using the *pygrank*[1] graph ranking library. We compare the following fairness-aware schemes on these two base algorithms:
None. The base ranking algorithm.
Mult. A simple postprocessing baseline that multiplies ranks across the sensitive and non-sensitive groups with a different constant each, so that disparate impact is fully mitigated. If r are the base ranking algorithm's node ranks, this method yields ranks:

$$r_{Mult}[v] = \left(\frac{\phi s[v]}{\sum_{u\in S} s[u]r[u]} + \frac{(1-\phi)(1-s[v])}{\sum_{u\notin S} s[u]r[u]}\right) r[v]$$

where $\phi = \frac{|S|}{|S|+|S'|}$ is the fraction of graph nodes that are sensitive and $s[u] = \{1 \text{ if } u \in S, 0 \text{ otherwise}\}$. It is easy to see that $\sum_{v\in S} r_{Mult}[v] = \sum_{v\notin S} r_{Mult}[v]$.
LFPRO. Near-optimal redistribution of ranks causing disparate impact [15].
Sweep. postprocessing using the sweep ratio of Equation 2.
FP. The FP model of Equation 5 whose probability parameters $a_S, a_{S'} \in [0, 1]$ and exponentias $b_S, b'_S \in [-10, 10]$ are trained the following objective with coordinate descent optimization provided by the *pygrank* library:

$$\text{minimize } \frac{\left\|L_\infty(r) - L_\infty(\hat{r}_{fair})\right\|_1}{|V|} - pRule(\hat{r}_{fair})$$

CFP. Constraining the FP model to not consider improvements for over 80% pRule by training it on the objective:

$$\text{minimize } \frac{\left\|L_\infty(r) - L_\infty(\hat{r}_{fair})\right\|_1}{|V|} - min\big(.8, pRule(\hat{r}_{fair})\big)$$

SweepLFPRO. Applying LFPRO on the outcome of Sweep.
SweepFP. Applying FP on the outcome of Sweep.
SweepCFP. Applying CFP on the outcome of Sweep.

[1] https://pypi.org/project/pygrank/

4.3 Evaluation

To compare the different fairness-aware methods, we randomly split graph nodes into training and evaluation sets, where the former comprise a fraction among $[10\%, 20\%, 30\%]$ of graph nodes, uniformly sampled without repetition. This mimics real-world usage of node ranking algorithms, where not many labels are known. For each possible fraction of training set nodes we sample 5 different training sets (we do so in a seeded way that passes the same sets to each ranking algorithm) and average the following measures across the respective $3 \cdot 5 = 15$ evaluation sets:

AUC. The area under curve of the receiver operating characteristics [35], which is often used to measure the quality of rank-based recommendations given known binary labels. 50% AUC indicates random node ranks, whereas 100% AUC perfect rank quality. We stress that fairness-aware methods are tasked with preserving but not improving potentially low node rank quality.

WR. The worst pRule between the ranks of all graph nodes and the ranks of evaluation nodes. To see why this measure is necessary, we point that some fairness-aware algorithms are designed to yield perfect disparate impact elimination (i.e. 100% pRule) when considering the whole graph. However, it is important for all nodes to benefit from increased fairness. For example, if a method achieves 100% and 1% pRule on all graph nodes and the evaluation ones respectively, it should not be considered as fair. For WR to accurately assess whether the effects of disparate impact treatment are uniformly spread across all nodes, we avoid directly optimizing towards the pRule of evaluation nodes.

In addition to the above-described evaluation, we also consider cases of extremely biased personalization. For example, this occurs with high probability when sensitive nodes are disproportionately few. Or when the personalization selection is biased against non-sensitive nodes, for example because corresponding people reluctant to share their information [36]. To simulate this behavior, we also perform experiments with the most extreme type of bias, where no biased node is allowed in the personalization.

5 Experiments

We first explore unbiased personalization, where training nodes are randomly selected before identifying queries of positive labels. The outcome of applying fairness-aware schemes on the personalized PageRank and Heat Kernel algorithms is detailed Table 2. We omit results for the Twitter graph, which by definition follows extreme personalization and is covered in subsequent experiments.

In this first series of experiments, base ranking algorithms comfortably exceed the baseline 80% pRule. Nevertheless, no method reaches perfect fairness for both all nodes and the evaluation subset. An important finding is that the Mult baseline outperforms LFRPO for producing fair ranks, which suggests that the latter's non-personalized efficacy does not carry over to personalization and uniform notions of fairness.

	Personalized Pagerank				Heat Kernels			
	Facebook0		Facebook686		Facebook0		Facebook686	
	AUC	WR	AUC	WR	AUC	WR	AUC	WR
None	.54	.90	.55	.92	.53	.85	.56	.83
Mult	.53	.95	.55	.94	.53	.89	.55	.85
LFRPO	.53	.94	.55	.92	.52	.81	.55	.74
Sweep	.55	.92	.58	.94	.54	.86	.58	.80
FP	.50	.94	.53	.96	.49	.95	.51	.96
CFP	.53	.92	.53	.91	.52	.89	.51	.92
SweepLFRPO	.54	.94	.57	.93	.53	.81	.57	.77
SweepFP	.56	.94	.55	.95	.56	.95	.54	.92
SweepCFP	.56	.95	.55	.91	.57	.88	.54	.88

Table 2. Experiments for unbiased personalization

Between approaches, our proposed FP and SweepFP dominate others with respect to WR values. SweepFP also maintains equal or better AUC compared to the base ranking algorithm. CFP and SweepCFP do not improve fairness as much, since their 80% pRule constraint is already satisfied, but still yield similar or higher rank quality and fairness compared to the base algorithms, Mult and LFPRO.

We now experiment on extreme personalization bias, where there is no sensitive group query node and fairness constrains require learning generalized rules that transfer to sensitive group nodes. Extreme bias ends up being too unfair for the base ranking algorithms to achieve 80% WR and they require further assistance. However, only methods that involve the FP model manage to uniformly mitigate disparate impact. These significantly reduce node rank quality compared to other approaches, however this happens because the latter fail to reach meaningful levels of fairness and hence settle on preserving the base node rank quality.

An interesting finding is that Sweep detrimentally affects FP on the Twitter graph. This indicates that either stationary bias is not exhibited by our base node ranking algorithms or that the success of Sweep for unbiased personalization in other cases can be attributed to its higher node rank quality proving more leeway for the FP model to improve rank and fairness trade-offs.

Overall, SweepFP achieves similar or better levels of uniform disparate impact mitigation and node rank quality trde-offs compared to other methods and is significantly ourperformed only by FP on the Twitter graph. We hence suggest using one of these two method when disparate impact mitigation is the most important objective of node ranking algorithms.

On the other hand, SweepCFP and CFP always achieve their constraint of reaching 80% WR while maintaining rank quality. This reveals that the FP model successfully prevents overfitting towards non-uniform notions of fairness. Hence, these methods should be preferred when ranking needs to satisfy only a predetermined fairness level.

	Personalized Pagerank						Heat Kernels					
	Facebook0		Facebook686		Twitter		Facebook0		Facebook686		Twitter	
	AUC	WR	AUC	WR	AUC	WR	AUC	WR	AUC	WR	AUC	WR
None	.53	.69	.54	.74	.58	0	.54	.37	.55	.39	.58	0
Mult	.51	.75	.52	.73	.49	.25	.50	.40	.52	.38	.56	.11
LFRPO	.51	.75	.52	.70	.54	.53	.42	.48	.50	.48	.57	.53
Sweep	.55	.67	.56	.74	.58	0	.54	.35	.56	.38	.58	0
FP	.53	.95	.52	.93	.49	.93	.47	.81	.52	.90	.44	.96
CFP	.52	.90	.52	.82	.53	.80	.49	.82	.52	.81	.45	.80
SweepLFRPO	.51	.72	.52	.68	.55	.52	.44	.44	.49	.43	.58	.53
SweepFP	.54	.91	.54	.92	.27	.96	.52	.78	.52	.80	.38	.94
SweepCFP	.54	.88	.54	.84	.43	.80	.53	.80	.54	.83	.48	.81

Table 3. Experiments for extreme personalization bias

The broader success of FP-based methods can be attributed to the preprocessing nature of personalization editing, which addresses the catastrophic effects of bias before it is propagated through complex network dynamics.

6 Conclusions and Future Work

In this work we tackled the problem of mitigating disparate impact while preserving the quality of graph node ranks and explored personalization editing as a means to do so. Our approach derives a personalization editing model whose parameters can be adjusted to trade-off rank preservation and fairness objectives. We explored this model's effectiveness on mitigating disparate impact while preserving the node rank quality of personalized PageRank and Heat Kernels by experimenting on three real-world social graphs, where found that it significantly outperforms potentially competing methods in uniformly mitigating bias across ranks, even under cases of extreme unfairness.

For future work, we are interested in exploring the efficacy of our methodology on more graphs and node ranking algorithms, especially ones that cannot be modeled as graph filters. Furthermore, the FP model or an adjustment could be used to mitigating other types of unfairness, such as disparate mistreatment and methods for mitigating unfairness under partial knowledge of sensitive attributes could be explored.

Acknowledgements

This work was partially funded by the European Commission under contract numbers H2020-860630 NoBIAS and H2020-825585 HELIOS.

References

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

2. J. Kleinberg, J. Ludwig, S. Mullainathan, and A. Rambachan, “Algorithmic fairness,” in *Aea papers and proceedings*, vol. 108, pp. 22–27, 2018.
3. E. Krasanakis, E. Spyromitros-Xioufis, S. Papadopoulos, and Y. Kompatsiaris, “Adaptive sensitive reweighting to mitigate bias in fairness-aware classification,” in *Proceedings of the 2018 World Wide Web Conference*, pp. 853–862, 2018.
4. M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi, “Fairness constraints: A flexible approach for fair classification.,” *J. Mach. Learn. Res.*, vol. 20, no. 75, pp. 1–42, 2019.
5. E. Ntoutsi, P. Fafalios, U. Gadiraju, V. Iosifidis, W. Nejdl, M.-E. Vidal, S. Ruggieri, F. Turini, S. Papadopoulos, E. Krasanakis, *et al.*, “Bias in data-driven artificial intelligence systems—an introductory survey,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 3, p. e1356, 2020.
6. D. Biddle, *Adverse impact and test validation: A practitioner’s guide to valid and defensible employment testing.* Gower Publishing, Ltd., 2006.
7. T. Calders and S. Verwer, “Three naive bayes approaches for discrimination-free classification,” *Data Mining and Knowledge Discovery*, vol. 21, no. 2, pp. 277–292, 2010.
8. F. Kamiran and T. Calders, “Data preprocessing techniques for classification without discrimination,” *Knowledge and Information Systems*, vol. 33, no. 1, pp. 1–33, 2012.
9. M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian, “Certifying and removing disparate impact,” in *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 259–268, 2015.
10. T. H. Haveliwala, “Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search,” *IEEE transactions on knowledge and data engineering*, vol. 15, no. 4, pp. 784–796, 2003.
11. L. Xiang, Q. Yuan, S. Zhao, L. Chen, X. Zhang, Q. Yang, and J. Sun, “Temporal recommendation on graphs via long-and short-term preference fusion,” in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 723–732, 2010.
12. J. Yang and J. Leskovec, “Defining and evaluating network communities based on ground-truth,” *Knowledge and Information Systems*, vol. 42, no. 1, pp. 181–213, 2015.
13. Y. Ding, E. Yan, A. Frazho, and J. Caverlee, “Pagerank for ranking authors in co-citation networks,” *Journal of the American Society for Information Science and Technology*, vol. 60, no. 11, pp. 2229–2243, 2009.
14. J. Heidemann, M. Klier, and F. Probst, “Identifying key users in online social networks: A pagerank based approach,” 2010.
15. S. Tsioutsiouliklis, E. Pitoura, P. Tsaparas, I. Kleftakis, and N. Mamoulis, “Fairness-aware link analysis,” *arXiv preprint arXiv:2005.14431*, 2020.
16. L. Page, S. Brin, R. Motwani, and T. Winograd, “The pagerank citation ranking: Bringing order to the web.,” tech. rep., Stanford InfoLab, 1999.
17. S. Segarra, A. G. Marques, and A. Ribeiro, “Optimal graph-filter design and applications to distributed linear network operators,” *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 4117–4131, 2017.
18. A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, “Graph signal processing: Overview, challenges, and applications,” *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

19. R. Andersen, F. Chung, and K. Lang, "Local partitioning for directed graphs using pagerank," in *International Workshop on Algorithms and Models for the Web-Graph*, pp. 166–178, Springer, 2007.
20. B. Bahmani, A. Chowdhury, and A. Goel, "Fast incremental and personalized pagerank," *Proceedings of the VLDB Endowment*, vol. 4, no. 3, 2010.
21. K. Kloster and D. F. Gleich, "Heat kernel based community detection," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1386–1395, 2014.
22. R. Andersen, F. Chung, and K. Lang, "Local graph partitioning using pagerank vectors," in *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pp. 475–486, IEEE, 2006.
23. R. Andersen, F. Chung, and K. Lang, "Using pagerank to locally partition a graph," *Internet Mathematics*, vol. 4, no. 1, pp. 35–64, 2007.
24. D. Chalupa, "A memetic algorithm for the minimum conductance graph partitioning problem," *arXiv preprint arXiv:1704.02854*, 2017.
25. X.-M. Wu, Z. Li, A. M. So, J. Wright, and S.-F. Chang, "Learning with partially absorbing random walks," in *Advances in neural information processing systems*, pp. 3077–3085, 2012.
26. A. Beutel, J. Chen, T. Doshi, H. Qian, L. Wei, Y. Wu, L. Heldt, Z. Zhao, L. Hong, E. H. Chi, *et al.*, "Fairness in recommendation ranking through pairwise comparisons," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2212–2220, 2019.
27. A. J. Biega, K. P. Gummadi, and G. Weikum, "Equity of attention: Amortizing individual fairness in rankings," in *The 41st international acm sigir conference on research & development in information retrieval*, pp. 405–414, 2018.
28. K. Yang and J. Stoyanovich, "Measuring fairness in ranked outputs," in *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, pp. 1–6, 2017.
29. M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates, "Fa* ir: A fair top-k ranking algorithm," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1569–1578, 2017.
30. A. Bose and W. Hamilton, "Compositional fairness constraints for graph embeddings," in *International Conference on Machine Learning*, pp. 715–724, 2019.
31. T. A. Rahman, B. Surma, M. Backes, and Y. Zhang, "Fairwalk: Towards fair graph embedding.," in *IJCAI*, pp. 3289–3295, 2019.
32. E. Dai and S. Wang, "Fairgnn: Eliminating the discrimination in graph neural networks with limited sensitive attribute information," *arXiv preprint arXiv:2009.01454*, 2020.
33. J. Leskovec and J. J. Mcauley, "Learning to discover social circles in ego networks," in *Advances in neural information processing systems*, pp. 539–547, 2012.
34. R. A. Rossi and N. K. Ahmed, "The network data repository with interactive graph analytics and visualization," in *AAAI*, 2015.
35. A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.
36. C. Cassel and A. Bindman, "Risk, benefit, and fairness in a big data world," *Jama*, vol. 322, no. 2, pp. 105–106, 2019.