Link Recommendations for PageRank Fairness

Sotiris Tsioutsiouliklis
King.com Ltd
Sweden
s.tsioutsiouliklis@king.com

Evaggelia Pitoura
University of Ioannina
Greece
pitoura@cs.uoi.gr

Konstantinos Semertzidis
IBM Research Europe
Ireland
konstantinos.semertzidis1@ibm.com

Panayiotis Tsaparas
University of Ioannina
Greece
tsap@uoi.gr

ABSTRACT

Network algorithms play a critical role in various applications, such as recommendations, diffusion maximization, and web search. In this paper, we focus on the fairness of such algorithms and in particular of PageRank. PageRank fairness refers to a fair allocation of the PageRank weights among the nodes. We consider the effect of the network structure on PageRank fairness. Concretely, we provide analytical formulas for computing the effect of edge additions on fairness and for the conditions that an edge must satisfy so that its addition improves fairness. We also provide analytical formulas for evaluating the role of existing edges in fairness. We use our findings to propose efficient linear time link recommendation algorithms for maximizing fairness, and we evaluate them on real datasets. Our approach can be seen as an effort towards making the network itself fairer as opposed to making fairer the network algorithms, or their outputs.

CCS CONCEPTS

• **Information systems** → **Social networks**; **Data mining**.

KEYWORDS

PageRank, link recommendations, algorithmic fairness

ACM Reference Format:
Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Konstantinos Semertzidis, and Panayiotis Tsaparas. 2022. Link Recommendations for PageRank Fairness. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3485447.3512249

1 INTRODUCTION

Algorithmic systems that exploit large datasets are increasingly being used in decision making, a fact that has raised important concerns about the trustworthiness of these decisions. Algorithmic fairness aims at addressing such concerns [11, 16, 32]. As graph algorithms play a critical role in a variety of applications, such as in recommendations, diffusion maximization, and web search, there has been recent research in algorithmic fairness for graphs as well (see e.g., [24, 42] for tutorials), including ranking [28, 41], embeddings [5, 8] and clustering [27].

In this paper, we address fairness with respect to the relative importance of the nodes in a graph as this is measured by PageRank. PageRank (pr) assigns a score to each node v that signifies the importance of v in the network globally, whereas personalized PageRank (ppr) rooted at a specific node u assigns a score to each node v that captures the relative importance of v for u [6, 19].

We focus on group-based fairness, where nodes belong to groups based on the value of some protected attribute. For example, in a social, or, cooperation network where nodes correspond to individuals, the protected attribute may be age, gender, or religion. Previous research has shown that under certain conditions the results of both pr and ppr may be unfair in terms of the PageRank weights assigned to each group [13, 41].

For handling PageRank unfairness recent research has proposed to modify the PageRank algorithm [28, 41]. In this paper, we take a different approach. We aim at modifying the network through link recommendations so that the results of pr and ppr are fairer. By doing so, *the network itself becomes fairer* with respect to the relative importance of each group in the network. Our approach is also different from pre-processing approaches where the input of the algorithm is augmented for the duration of the algorithm [37, 43]. Instead, we propose augmentations towards making the data (the network in our case) fairer in the long run.

We provide analytical formulas for the effect of edge additions on pr and ppr and we derive the conditions that the endpoints of an edge must satisfy so that its addition improves fairness. We also provide formulas for evaluating the contribution on fairness of existing edges by measuring the impact that their removal has on fairness. In simple terms, edges appropriate for increasing fairness towards a group are edges that have sources of small degree and high PageRank value, and point to a node located in a network area where the group is less represented than in the area of the source.

Link recommendation algorithms play a central role in networks, since they control how a network grows over time [29, 30]. In this paper, we propose a link recommendation algorithm that suggests links for improving fairness. We present an efficient linear-time link recommendation algorithm that exploits absorbing random walks. We also present a hybrid algorithm that considers both

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France

ACM ISBN 978-1-4503-9096-5/22/04...$15.00
https://doi.org/10.1145/3485447.3512249

fairness and the probability that the link is accepted. We evaluate the effectiveness of our algorithms in terms of accuracy and fairness using real datasets.

In summary, in this paper, we make the following contributions:

- We provide analytical formulas for the effect of edge additions and deletions on PR and PPR fairness.
- We present an efficient link recommendation algorithm for PR and PPR fairness that exploits absorbing random walks.
- We report experimental results using real graphs that evaluate network edges and link recommendation algorithms in terms of PR and PPR fairness.

The rest of this paper is structured as follows. We first present related work in Section 2. In Section 3, we formally define fairness and the research questions addressed in this paper. In Section 4, we present our formulas for link fairness, and in Section 5 our link recommendation algorithms. Our experimental results are reported in Section 6, and conclusions in Section 7.

2 RELATED WORK

Algorithmic fairness has attracted a lot of research interest (e.g., see [17, 32, 35] for recent surveys). Lately, there has been also research effort in graph algorithms (e.g., see [24, 42] for recent tutorials), including group-based fairness for centrality measures [28, 41], embeddings [5, 8], influence maximization [15, 40] and clustering [27]. There are also individual fairness approaches based on the premise that similar nodes should be treated similarly [23].

In this paper, we focus on centrality and in particular on PageRankcentrality. To the best of our knowledge, this is the first approach to link recommendations for PageRank fairness.

Network centrality fairness. Previous research has studied network fairness in terms of degree centrality. It was shown that homophily, preferential attachment and discrepancies in the size of the groups may lead to a glass ceiling effect, i.e., the underrepresentation of the minority group in top degree positions [3]. It has also been shown that this effect can be exacerbated by recommendation algorithms [39] and that degree inequalities exist in real social networks [26]. Very recent research has also found inequities in the PageRank distributions between groups [13, 41].

To address PageRank unfairness, previous research has proposed modifying the inner-workings of the PageRank algorithm, so that the resulting algorithm is fair, and its output is as close as possible to the original PageRank[41]. The authors of [28] propose making personalized versions of ranking fair with minimal changes from the original ranks. In this paper, we do not modify PR or PPR, instead we modify the network through link recommendations so that the output of these algorithms on the modified network is fairer.

Fair link recommendations. Research on fairness in link recommendations looks at the presence of the minority group in the recommendation lists. A commonly used objective is demographic parity (termed *disparate visibility* in [14] and *equality of representation* in [36]) that asks that the percentage of the members of the minority group in the recommendation lists is equal to their percentage in the overall population. The authors of [36] propose a variation of the node2vec embedding [21] that uses a fair random walk to achieve equality of representation. A similar concept called *dyadic-level protection* is introduced in [31] to reduce homophily by promoting links that connect nodes belonging to different groups.

There have been recommendations also for other network properties, including improving closeness centrality [34], fighting opinion control [2] and reducing controversy and polarization [18, 22].

PageRank optimization. There has been some previous work in the context of web on strategies for increasing the PageRank of specific nodes. For example, it was shown that the optimal linking strategy for a node is to have only one outgoing link pointing to a node with the shortest mean first passage time back to it [4]. This result was generalized to provide an optimal linking strategy for increasing the PageRank of a given set of nodes [9]. The authors of [7] consider the problem of maximizing the PageRank of a node by selecting edges from a predefined set. Finally, the authors of [25] formulate the *PageRank auditing problem* of locating the k graph elements, e.g., edges, nodes, subgraphs, whose removal would result in the largest modification of the PageRank vector. These works aim at optimizing PageRank and do not address fairness.

3 DEFINITIONS

In this section, we introduce the main concepts necessary for our work and we define the problems we consider in this paper.

3.1 The PageRank Algorithm

The PageRank (PR) algorithm [6] pioneered link analysis for weighting and ranking the nodes of a graph. It was popularized by its use in the Google search engine, but it has found a wide range of applications in different settings [19]. The algorithm takes as input a directed graph $G = (V, E)$, and produces a scoring vector $\mathbf{p}$, that assigns a weight to each node $v \in V$. The scoring vector is the stationary distribution of a *random walk with restarts* on G. The transition matrix $\mathbf{P}$ of the random walk is defined as the normalized adjacency matrix of graph G. The algorithm is parameterized by the value γ, which is the restart probability, and the jump vector $\mathbf{v}$, which defines a distribution over the nodes in the graph, according to which the restart node is selected. Typically, $\gamma = 0.15$, and the jump vector is the uniform vector $\mathbf{u}$. For nodes with no outgoing edges, we adopt the convention that the random walk performs a jump to a node chosen uniformly at random [19]. The Pagerank vector $\mathbf{p}$ satisfies the equation:

$$\mathbf{p}^T = (1-\gamma)\mathbf{p}^T\mathbf{P} + \gamma\,\mathbf{v}^T \tag{1}$$

A special case of the PageRank algorithm is the Personalized PageRank (PPR) algorithm, where the restart vector is a unit vector $\mathbf{e}_i$ that puts all the probability mass on a single node i. We use $\mathbf{p}_i$ to denote the PPR vector for node i. We say that node i *allocates* pagerank $\mathbf{p}_i(u)$ to node u. Personalized PageRank provides a "view" of the network with respect to a specific node.

The following lemma will be useful in our analysis.

Lemma 3.1. *For the PageRank vector* $\mathbf{p}$*, it holds* $\mathbf{p}^T = \mathbf{v}^T\mathbf{Q}$*, where:*

(1) $\mathbf{Q} = \gamma\,(\mathbf{I} - (1-\gamma)\mathbf{P})^{-1}$.
(2) *The i-th column vector* $\mathbf{Q}_i$ *corresponds to the personalized PageRank vector of node i, that is:* $\mathbf{p}_i = \mathbf{Q}_i$.

Proof. We obtain (1) directly from Equation 1. For (2), if we set $\mathbf{v} = \mathbf{e}_i^T$, then $\mathbf{p} = \mathbf{Q}_i$, the i-th row of matrix $\mathbf{Q}$. □

Given Lemma 3.1, we will use interchangeably $\mathbf{p}_i$ and $\mathbf{Q}_i$ to denote the PPR vector for node i. The $\mathbf{Q}_{ij}$ entry of the matrix $\mathbf{Q}$ is the PPR weight $\mathbf{p}_i(j)$ that node i allocates to node j.

3.2 PageRank Fairness

We assume two *groups* of nodes, R and B, of red and blue nodes, defined based on some node attributes. Given a group S (either R or B), we use $ratio(S) = \frac{|S|}{|V|}$ to denote the ratio of group S in the overall population. Abusing the notation, we will use $\mathbf{p}(S)$ to denote the PageRank mass allocated to group S, that is $\mathbf{p}(S) = \sum_{i \in S} \mathbf{p}(i)$. We refer to $\mathbf{p}(S)$ as the PR ratio of group S.

Given a *target* group S, and a parameter ϕ, we say that the network is *PR-unfair* to group S, if $\mathbf{p}(S) < \phi$. Parameter ϕ is input to our definition. It can be specified so as to implement different fairness policies. We will assume as default, $\phi = ratio(S)$. This means that we ask that the ratio of the PageRank weights assigned to the group S is at least equal to the ratio of the group in the overall population, analogously to demographic parity [11].

Similarly, given a node v, we use $\mathbf{p}_v(S)$ to denote the personalized PageRank mass allocated to group S by node v, that is $\mathbf{p}_v(S) = \sum_{i \in S} \mathbf{p}_v(i)$. To define PPR-unfairness, as in [41], we exclude the probability mass γ allocated to node v through the random jump so as to consider only the fraction of the organic PPR that is allocated to group S. Specifically, for a node v, we define $\overline{\mathbf{p}_v}(S) = \frac{\mathbf{p}_v(S) - \gamma \mathbb{1}(v \in S)}{1-\gamma}$, where $\mathbb{1}(v \in S)$ is an indicator function that is 1 if $v \in S$. Given the target group S and a parameter ϕ, we say that node v is *PPR-unfair* to group S, if $\overline{\mathbf{p}_v}(S) < \phi$.

Note that for $\phi = ratio(S)$, if $\overline{\mathbf{p}_v}(S) \geq \phi$ for all $v \in V$, then $\mathbf{p}(S) \geq \phi$. That is, if all nodes are PPR-fair to S, then the network is PR-fair to S. The opposite is not always true. The proof of this property appears in the Appendix.

Given group S, we measure PageRank fairness (PR fairness) by the PR ratio $\mathbf{p}(S)$. Similarly, for a node v, we measure personalized PageRank fairness (PPR fairness) by the PPR ratio $\mathbf{p}_v(S)$. Intuitively, PR fairness provides a global, or network-level view of fairness, while PPR fairness a local, or per-node view of fairness. In this work, we consider the problem of increasing the PR and PPR fairness for a group S by modifying the underlying structure of the graph G. We address the following research questions in this direction.

What is the effect of edge additions on fairness? We derive analytical formulas for estimating the change in the PR and PPR ratios for the target group S, when adding a single edge (x, y), as well as when adding a set of edges to a node x in the graph.

What is the contribution of an existing edge to fairness? We derive analytical formulas for estimating the contribution of an edge $(x, y) \in G$ to the PR and PPR fairness for the target group S.

What edges should we recommend to a user to increase fairness? We propose efficient algorithms for finding the best k edges to recommend to a node x so as to maximize the increase in the PR, or PPR ratio for the target group S.

4 THE ROLE OF LINKS IN FAIRNESS

In this section, we focus on the role that links play in fairness. We provide a closed-form formula for the effect of edge additions on fairness, and we prove a necessary and sufficient condition that an edge must satisfy so that its addition results in increasing fairness. Finally, we analyze the role of existing edges in fairness.

4.1 Fairness Gain by Adding Links

We will now compute the gain in fairness of adding a single edge (x, y). Let $G = (V, E)$ denote the underlying graph of the network, and let (x, y) be an edge not in G. Let $G' = (V, E \cup \{(x, y)\})$ denote the network after the addition of the edge (x, y), and let $\mathbf{p}'$ and $\mathbf{p}'_u$ denote the PR and PPR vectors on graph G'. We define the *fairness gain* for group S (either R or B) of the addition of the edge (x, y), as $fgain((x, y), S) = \mathbf{p}'(S) - \mathbf{p}(S)$, that is, the change in the PR ratio of group S, when adding the edge (x, y). Similarly, for a node u we define the *personalized fairness gain* for group S, $pgain_u((x, y), S) = \mathbf{p}'_u(S) - \mathbf{p}_u(S)$, that is the change in the PPR ratio. Note that the value of $fgain$ and $pgain$ may be negative for some edges.

The following theorem estimates analytically the gains. For a node x, we use d_x to denote the out-degree of the node, and N_x to denote the out-neighbors of the node.

Theorem 4.1. *Let $G = (V, E)$ be a graph, S the target group, and $(x, y) \notin E$ an edge not in G. Let*

$$\Lambda((x,y),S) = \begin{cases} \dfrac{\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(S) - \frac{1}{d_x}\sum_{w \in N_x}\mathbf{p}_w(S)\right)}{d_x + 1 - \frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(x) - \frac{1}{d_x}\sum_{w \in N_x}\mathbf{p}_w(x)\right)}, & d_x \neq 0 \\ \dfrac{\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(S) - \frac{1}{|V|}\sum_{w \in V}\mathbf{p}_w(S)\right)}{1 - \frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(x) - \frac{1}{|V|}\sum_{w \in V}\mathbf{p}_w(x)\right)}, & d_x = 0 \end{cases} \tag{2}$$

(1) *The fairness gain for group S of adding the edge (x, y) to G is:*

$$fgain((x,y),S) = \Lambda((x,y),S)\,\mathbf{p}(x) \tag{3}$$

(2) *The personalized fairness gain for node u for group S of adding the edge (x, y) to G is:*

$$pgain_u((x,y),S) = \Lambda((x,y),S)\,\mathbf{p}_u(x) \tag{4}$$

Proof. Let $\mathbf{P}$ and $\mathbf{P}'$ denote the transition matrices of the PageRank random walk on the graphs G and G' before and after the addition of the edge (x, y) respectively. To prove our theorem, we first write the transition matrix $\mathbf{P}'$ as the sum of the transition matrix $\mathbf{P}$ and a rank-1, perturbation matrix $\mathbf{D}$. For the following we assume that $d_x \neq 0$. $\mathbf{D}_i$ denotes the i-th row of matrix $\mathbf{D}$.

$$\mathbf{P}' = \mathbf{P} + \mathbf{D}, \quad \mathbf{D}_i = \begin{cases} 0, & i \neq x \\ (-\frac{1}{d_x+1})\mathbf{P}_x + \frac{1}{d_x+1}\mathbf{e}_y^T, & i = x \end{cases}$$

where $\mathbf{e}_y$ is the vector with 1 at position y and 0 everywhere else. We want to estimate

$$\mathbf{Q}' = \gamma\left(\mathbf{I} - (1-\gamma)\mathbf{P}'\right)^{-1} = \gamma\left(\mathbf{I} - (1-\gamma)(\mathbf{P} + \mathbf{D})\right)^{-1}.$$

To do so, we exploit a fundamental lemma [33] that states that for a non-singular matrix $\mathbf{M}$ and a rank-1 matrix $\mathbf{H}$, such that $\mathbf{M} + \mathbf{H}$ is nonsingular, we have:

$$(\mathbf{M} + \mathbf{H})^{-1} = \mathbf{M}^{-1} - \frac{1}{1+g}\mathbf{M}^{-1}\mathbf{H}\mathbf{M}^{-1}, \; g := tr(\mathbf{H}\mathbf{M}^{-1})$$

Applying for $\mathbf{M} = (\mathbf{I} - (1-\gamma)\,\mathbf{P})$ and $\mathbf{H} = -(1-\gamma)\,\mathbf{D}$, and using the fact that $\mathbf{Q} = \gamma\mathbf{M}^{-1}$:

$$\begin{aligned}
\mathbf{Q}' &= \gamma(\mathbf{M}+\mathbf{H})^{-1} \\
&= \gamma\mathbf{M}^{-1} - \gamma\frac{1}{1+g}\mathbf{M}^{-1}\mathbf{H}\mathbf{M}^{-1},\ g := tr(\mathbf{H}\mathbf{M}^{-1}) \\
&= \gamma\frac{\mathbf{Q}}{\gamma} - \gamma\frac{1}{1+h}\ \frac{\mathbf{Q}}{\gamma}\left(-(1-\gamma)\cdot\mathbf{D}\right)\frac{\mathbf{Q}}{\gamma},\ h := tr\left(-(1-\gamma)\mathbf{D}\frac{1}{\gamma}\mathbf{Q}\right) \\
&= \mathbf{Q} + \frac{\frac{(1-\gamma)}{\gamma}}{1-\frac{(1-\gamma)}{\gamma}q}\mathbf{Q}\,\mathbf{D}\,\mathbf{Q},\ \text{where}\ \ q := tr(\mathbf{D}\,\mathbf{Q}) \qquad (5)
\end{aligned}$$

With mathematical manipulations, we get:

$$\mathbf{DQ}_{ij} = \begin{cases} 0, & i \neq x \\ \frac{1}{d_x+1}\left(\mathbf{Q}_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wj}\right), & i = x \end{cases}$$

$$\mathbf{QDQ}_{ij} = \frac{1}{d_x+1}\mathbf{Q}_{ix}\left(\mathbf{Q}_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wj}\right)$$

Substituting in Equation 5, and using the fact that $q = tr(\mathbf{D}\,\mathbf{Q}) = \mathbf{DQ}_{xx}$ we have:

$$\mathbf{Q}'_{ij} = \mathbf{Q}_{ij} + \mathbf{Q}_{ix}\frac{\frac{(1-\gamma)}{\gamma}\left(\mathbf{Q}_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wj}\right)}{d_x + 1 - \frac{(1-\gamma)}{\gamma}\left(\mathbf{Q}_{yx} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wx}\right)} \qquad (6)$$

From Lemma 3.1, we have that $\mathbf{p}_i(x) = \mathbf{Q}_{ix}$ and $\mathbf{p}'_i(S) = \sum_{j\in S}\mathbf{Q}'_{ij}$. Summing Equation 6 over $j \in S$:

$$\begin{aligned}
\mathbf{p}'_i(S) &= \sum_{j\in S}\mathbf{Q}_{ij} + \mathbf{Q}_{ix}\frac{\frac{(1-\gamma)}{\gamma}\left(\sum_{j\in S}\mathbf{Q}_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\sum_{j\in S}\mathbf{Q}_{wj}\right)}{d_x+1-\frac{(1-\gamma)}{\gamma}\left(\mathbf{Q}_{yx} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wx}\right)} \\
&= \mathbf{p}_i(S) + \mathbf{p}_i(x)\frac{\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(S) - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{p}_w(S)\right)}{d_x+1-\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(x) - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{p}_w(x)\right)} \\
&= \mathbf{p}_i(S) + \mathbf{p}_i(x)\Lambda((x,y),S) \qquad (7)
\end{aligned}$$

Applying Equation 7 for $i = u$, we obtain Equation 4 for $pgain((x,y),S) = \mathbf{p}'_i(S) - \mathbf{p}_i(S)$.

Using the fact that $\mathbf{p}'(S) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{p}'_i(S)$ and $\mathbf{p}(x) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i(x)$:

$$\mathbf{p}'(S) = \mathbf{p}(S) + \mathbf{p}(x)\Lambda((x,y),S) \qquad (8)$$

Subtracting, we get Equation 3 for $fgain((x,y),S) = \mathbf{p}'(S) - \mathbf{p}(S)$.

The formula for the case where $d_x = 0$ follows from the fact that the $\mathbf{P}_x$ vector in the definition of matrix $\mathbf{D}$ is the uniform vector $\mathbf{u}$ with transition probability $1/|V|$ to all nodes in the graph. □

Theorem 4.1 formulates the following intuitive observations. Regarding the *source node* x of the edge to be added, the PR fairness gain is proportional to its PR since the PR of x is the quantity to be distributed to the nodes in S through the new edge. The higher this quantity, the stronger the effect of the edge. Correspondingly, the PPR gain for a node u is proportional to the PPR of u that is allocated to x, since again, this is the quantity to be distributed. Note that adding an edge whose source node x is not reachable from u (i.e., $\mathbf{p}_u(x) = 0$) has no effect on the PPR fairness of u. The gain is also inversely proportional to the out-degree of x, since the smaller the degree, the largest the PR (PPR) portion of x that will go to y. Thus, the source nodes that affect fairness the most are central nodes with small out-degree.

Regarding *the target node* y, good target nodes are nodes whose PPR-ratio $\mathbf{p}_y(S)$ is larger than the average PPR-ratio of the current neighbors of x. Intuitively, $\mathbf{p}_y(S)$ is roughly the fraction of the PR that reaches y that will end up to nodes in S. The higher this is, the stronger the effect of the new edge. However, the new edge takes away some PR from the existing neighbors of x. It pays off to add the new edge only if it is better than the existing edges on average. Intuitively, this means that we should prefer to connect with nodes that favor S more that the current neighbors of the source node.

Lastly, the *quantity in the denominator* accounts for the difference between the PR that the target node y gives to the source node x, and the average PR that the current neighbors of x give to x. We can think of the PPR $\mathbf{p}_w(x)$ for a neighbor w of x as the return probability to x. The higher it is, the faster we close the loop to x. Loops boost PageRank, and thus increase the overall gain. Since again new links act competitively to existing ones, we want the new edge to close the loop faster than the existing edges on average. Ideally, we want to connect x to a node y that already points to x.

The following corollaries are direct implications of Theorem 4.1:

Corollary 4.2. *An edge (x, y) whose addition increases the PPR ratio $\mathbf{p}_u(S)$ of a node u, increases the PPR ratio of all nodes $v \in V$ in the network, for which there is a path to node x.*

Corollary 4.3. *Given a node $x \in G$, an edge (x, y) maximizes the fairness gain $fgain((x,y),S)$ if and only if it maximizes the personalized fairness gain $pgain_u((x,y),S)$, $u \in V$.*

We also provide necessary and sufficient conditions for the gain to be positive. We can show that an edge (x, y) increases both the PR and PPR ratio for group S, if and only if, the PPR ratio of the target node y is larger than the average PPR ratio of the current neighbors of the source node x. The proof of the following Lemma appears in the Appendix, and relies on the fact that we can prove that the denominator in the formula for $\Lambda((x,y),S)$ is always positive.

Lemma 4.4. *Let $G = (V, E)$ be a graph. Adding edge (x, y) to G increases the PR ratio $\mathbf{p}(S)$ and the PPR ratios $\mathbf{p}_u(S)$ for the target group S, if and only if:*

$$\mathbf{p}_y(S) > \frac{1}{d_x}\sum_{w\in N_x}\mathbf{p}_w(S)$$

Adding a Set of Edges. Theorem 4.1 can be generalized for the case of adding a set of k edges, $k > 1$, to a source node x. The proof of the following Theorem appears in the Appendix.

Theorem 4.5. *Let $G = (V, E)$ be a graph, S the target group, x a node in V, $E_x = \{(x, y_i) \notin E, i = 1, \ldots, k\}$ a set of k edges not in G with source x and V_y the set of the endpoints of these edges. Let*

$$\Lambda(E_x, S) = \begin{cases} \dfrac{\frac{1-\gamma}{\gamma}\left(\frac{1}{k}\sum_{y\in V_y}\mathbf{p}_y(S) - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{p}_w(S)\right)}{\frac{d_x+k}{k} - \frac{1-\gamma}{\gamma}\left(\frac{1}{k}\sum_{y\in V_y}\mathbf{p}_y(x) - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{p}_w(x)\right)}, & d_x \neq 0 \\ \dfrac{\frac{1-\gamma}{\gamma}\left(\frac{1}{k}\sum_{y\in V_y}\mathbf{p}_y(S) - \frac{1}{|V|}\sum_{w\in V}\mathbf{p}_w(S)\right)}{1 - \frac{1-\gamma}{\gamma}\left(\frac{1}{k}\sum_{y\in V_y}\mathbf{p}_y(x) - \frac{1}{|V|}\sum_{w\in V}\mathbf{p}_w(x)\right)}, & d_x = 0 \end{cases}$$

(1) *The fairness gain of adding the set of edges E_x to G for group S is $fgain(E_x, S) = \Lambda(E_x, S)\ \mathbf{p}(x)$.*
(2) *The personalized fairness gain for node u for group S of adding the set of edges E_x to G is $pgain_u(E_x, S) = \Lambda(E_x, S)\ \mathbf{p}_u(x)$.*

4.2 Fairness Importance of Existing Links

It is also interesting to understand the role that an existing edge (x, y) plays in the fairness of a network. We do so by considering the effect of removing the specific edge from the network on PageRank fairness. Given a graph $G = (V, E)$, and an edge $(x, y) \in E$, we define $G' = (V, E \setminus \{(x, y)\})$ to be the graph after the removal of edge (x, y), and $\mathbf{p}'$ and $\mathbf{p}'_i$ the corresponding PR and PPR vectors. For group S, we use $fvalue((x, y), S) = \mathbf{p}(S) - \mathbf{p}'(S)$ and $pvalue_u((x, y), S) = \mathbf{p}_u(S) - \mathbf{p}'_u(S)$, for measuring the contribution of the edge (x, y) to the PR fairness of the network to group S, and the PPR fairness of a specific node u to group S respectively. The proof of the following Theorem appears in the Appendix.

THEOREM 4.6. *Let $G = (V, E)$ be a directed graph, S the target group, and $(x, y) \in E$ a (directed) edge in G. Let*

$$\Lambda_D((x, y), S) = \begin{cases} \dfrac{\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(S) - \frac{1}{d_x}\sum_{w \in N_x} \mathbf{p}_w(S)\right)}{d_x - 1 - \frac{1-\gamma}{\gamma}\left(\frac{1}{d_x}\sum_{w \in N_x} \mathbf{p}_w(x) - \mathbf{p}_y(x)\right)}, & d_x > 1 \\ \dfrac{\frac{1-\gamma}{\gamma}\left(\mathbf{p}_y(S) - \frac{1}{|V|}\sum_{w \in V} \mathbf{p}_w(S)\right)}{1 - \frac{1-\gamma}{\gamma}\left(\frac{1}{|V|}\sum_{w \in V} \mathbf{p}_w(x) - \mathbf{p}_y(x)\right)}, & d_x = 1 \end{cases}$$

(1) *The fairness value of edge (x, y) for group S is $fvalue((x, y), S) = \Lambda_D((x, y), S)\ \mathbf{p}(x)$.*
(2) *The personalized fairness value of edge (x, y) for node $u \in V$ for group S is $pvalue_u((x, y), S) = \Lambda_D((x, y), S)\ \mathbf{p}_u(x)$.*

5 LINK RECOMMENDATIONS FOR FAIRNESS

In this section, we present our link recommendation algorithms that recommend edges so as to increase the PR or the PPR fairness of the target group S.

5.1 Recommending a Single Edge

To recommend a single link to a given source node x, we use Theorem 4.1 to compute the fairness gain ($fgain$, or $pgain$) for each candidate edge, and then select the one with the highest gain. A straightforward way to apply the theorem is to first compute the PPR vectors for all nodes in the graph, and then use Equation 2 to compute $\Lambda((x, y), S)$ for each candidate edge (x, y). The edge with the highest Λ value is the one with the highest gain. This requires $O(|V|)$ PageRank computations, resulting in overall $O(|V|^2 + |V||E|)$ time.

We present a more efficient algorithm for selecting the best edge to recommend, the BFE algorithm, shown in Algorithm 1. The BFE algorithm has two main steps: one for computing the PPR ratio $\mathbf{p}_v(S)$ for all nodes v (lines 1-3), and one for computing the PPR values $\mathbf{p}_v(x)$ for all nodes v (lines 4-6). We will show that these steps can be implemented with just two PageRank-like iterative computations, resulting in overall complexity $O(|V| + |E|)$.

The efficient computation of the BFE algorithm relies on the use of *absorbing random walks* [10, 20]. In an absorbing random walk, we have two types of nodes: *transient* nodes, from which we transition as in a regular random walk, and *absorbing* or sink nodes, out of which we cannot transition, and thus the walk is *absorbed*. For an absorbing random walk with τ transient nodes and α absorbing nodes, the transition matrix $\mathbf{N}$ of the random walk

Algorithm 1 Best Fair Edge (BFE) Algorithm

Require: Graph $G(V, E)$, source node $x \in V$, group S
1: **for** each $v \in V$ **do**
2: Compute $p_v(S)$
3: **end for**
4: **for** each $v \in V$ **do**
5: Compute $p_v(x)$
6: **end for**
7: **return** $\arg\max_v\ gain(x, v)$

is in the form:

$$\mathbf{N} = \begin{bmatrix} \mathbf{T} & \mathbf{R} \\ \mathbf{0}_{\alpha \times \tau} & \mathbf{I}_\alpha \end{bmatrix}$$

Matrix $\mathbf{T}$ is the $\tau \times \tau$ transition matrix between transient nodes, while matrix $\mathbf{R}$ is the $\tau \times \alpha$ transition matrix from the transient to the absorbing nodes. There are no transitions from absorbing nodes to transient nodes and each absorbing node loops back to itself.

A useful matrix in absorbing random walks is the $\tau \times \alpha$ matrix $\mathbf{B}$ with the *absorption probabilities*: $\mathbf{B}_{ij}$ is the probability that a random walk that starts from transient node i will be absorbed at absorbing node j. We can compute $\mathbf{B}$ using the *fundamental* matrix $\mathbf{F}$ of the absorbing random walk. The fundamental matrix $\mathbf{F}$ is a $\tau \times \tau$ matrix, where $\mathbf{F}_{ij}$ is the expected number of times that a random walk that starts from transient node i is in transient node j after an infinite number of steps. It holds that $\mathbf{F} = (\mathbf{I} - \mathbf{T})^{-1}$, and $\mathbf{B} = \mathbf{F}\mathbf{R}$ [20].

For an absorbing node a, the computation of $\mathbf{B}_{ia}$ can be done through an iterative algorithm. Node a has absorption probability 1 of being absorbed at itself, while the other absorbing nodes have probability 0 of being absorbed at a. Initially, all transient nodes have probability 0 of being absorbed at node a. At each iteration, each transient node updates its absorption probability as the average of the absorption probabilities of its neighbors [10]. The process is repeated until convergence. The computation is very similar to that of PageRank and it can be done in time $O(|V| + |E|)$.

Computing the $\mathbf{p}_v(x)$ vector: We first show how to use absorbing random walks to perform the computation in lines 4-6. We define an absorbing random walk $\overline{X}$ as follows. Given the graph G, we add two adsorbing nodes a_x and a_o, and we connect node x to node a_x and all other nodes to node a_o, all with probability γ. The transition matrix $\overline{\mathbf{N}}$ of $\overline{X}$ is:

$$\overline{\mathbf{N}} = \begin{bmatrix} (1-\gamma)\mathbf{P} & \overline{\mathbf{R}} \\ \mathbf{0}_{2xn} & \mathbf{I}_2 \end{bmatrix}, \quad \overline{\mathbf{R}} \in \mathbb{R}^{nx2},$$

where $\mathbf{P}$ is the transition matrix of the PageRank random walk, and

$$\overline{\mathbf{R}}_{ij} = \begin{cases} \gamma, & i = x, j = a_x, \text{and } i \neq x, j = a_o \\ 0, & \text{otherwise} \end{cases}$$

We can now see the connection between the absorbing random walk $\overline{X}$ and PageRank. Let $\overline{\mathbf{F}}$ denote the fundamental matrix of $\overline{\mathbf{X}}$. It holds that $\overline{\mathbf{F}} = (\mathbf{I} - (1-\gamma)\mathbf{P})^{-1}$ and thus, $\overline{\mathbf{F}} = \frac{\mathbf{Q}}{\gamma}$. Let $\overline{\mathbf{B}}$ be the absorption probability matrix of $\overline{\mathbf{X}}$. We prove the following:

LEMMA 5.1. *The personalized PageRank of node i to node x is equal to the absorption probability of node i to node a_x: $\mathbf{p}_i(x) = \overline{\mathbf{B}}_{ia_x}$.*

Proof. We have that $\overline{\mathbf{B}} = \overline{\mathbf{F}}\,\overline{\mathbf{R}} = \frac{1}{\gamma}\mathbf{Q}\,\overline{\mathbf{R}}$. Therefore,

$$\overline{\mathbf{B}}_{ia_x} = \frac{1}{\gamma}\sum_{k\in V}\mathbf{Q}_{ik}\overline{\mathbf{R}}_{ka_x} = \frac{1}{\gamma}\mathbf{Q}_{ix}\gamma = \mathbf{Q}_{ix}$$

□

Given the efficient computation of the $\overline{\mathbf{B}}_{ia_x}$ probabilities, we can compute the PPR values of all nodes for node x in time $O(|V| + |E|)$.

Computing the PPR ratio $\mathbf{p}_i(S)$: We now show how to use absorbing random walks for the computation in lines 1-3. We define an absorbing random walk $\widetilde{X}$ as follows. Given the graph G, we add two absorbing nodes a_r and a_b, representing the red and the blue group respectively. We add an edge from each red node to node a_r with probability γ, and an edge from each blue node to node a_b with probability γ. Let $\widetilde{\mathbf{B}}_{ia_r}, \widetilde{\mathbf{B}}_{ia_b}$ denote the absorption probabilities for node i to a_r and a_b respectively. The proof of the following Lemma is similar to that of Lemma 5.1.

Lemma 5.2. *The PPR ratio of node i for groups R and B is equal to the absorption probability of node i to node a_r and a_b respectively:* $\mathbf{p}_i(R) = \widetilde{\mathbf{B}}_{ia_r}$, *and* $\mathbf{p}_i(B) = \widetilde{\mathbf{B}}_{ia_b}$.

Working with $\widetilde{X}$ as before we can compute the PPR ratio of all nodes for the target group S in time $O(|V| + |E|)$.

Putting it all together, the BFE algorithm requires only two PageRank-like computations to compute the gain for all candidate edges, resulting in $O(|V| + |E|)$ complexity.

5.2 Recommending More than one Link

We now consider the case where we recommend multiple links to a source node. We adopt a greedy algorithm for the problem that constructs the set k of edges to recommend iteratively, each time adding the edge that incurs the maximum fairness gain when added to the set. Specifically, at each iteration, given the set L of the edges selected so far, for a candidate edge (x, y), the algorithm estimates the incremental fairness gain $fdelta(L, (x, y)) = gain(L \cup \{(x, y)\}, S) - gain(L, S)$ of adding edge (x, y) to the graph, and adds the edge with the maximum $fdelta$ to the set. The $gain$ may be either the PR fairness gain ($fgain$), or the PPR fairness gain ($pgain$).

A naive implementation of the greedy algorithm would create the graph $G_L = (V, E \cup L)$ at each iteration, and estimate the gain for each candidate edge (x, y) on G_L using the BFE algorithm. This requires $O(k(|V| + |E|))$ time. We improve the efficiency of the algorithm by exploiting Theorem 4.5. Note that for a graph G and a set of edges L, the computation of $\Lambda(L, S)$ utilizes the $\mathbf{p}_i(x)$, and $\mathbf{p}_i(S)$ values computed *on the original graph G*. We can thus compute these values once and use them to estimate $fdelta(L, (x, y))$ in constant time. Using absorbing random walks, we can compute these quantities in time $O(|V| + |E|)$ and therefore, the complexity of the greedy algorithm is $O(k|V| + |E|)$. The outline of the algorithm is shown in the Appendix.

Given this generic Greedy algorithm, we define two algorithms:

• The FREC algorithm which, given a node x and the group S, looks for the set of edges $L = \{(x, y) : y \notin G\}$ that maximizes the PR fairness gain $fgain(L, S)$ for the group S.

• The PREC algorithm which, given a node x and the group S, looks for the set of edges $L = \{(x, y) : y \notin G\}$ that maximizes the PPR fairness gain $pgain_x(L, S)$ for the group S.

Table 1: Dataset characteristics.

Dataset	#nodes	#edges	ratio(R)	red PR	h	Protected attr. (R)
BOOKS	92	748	0.467	0.474	0.065	political (left)
BLOGS	1,222	16,717	0.485	0.332	0.169	political (left)
DBLP-GEN	16,501	66,613	0.257	0.249	0.898	gender (women)
DBLP-PUB	16,501	66,613	0.080	0.061	0.723	pub-year (≥ 2016)
TWITTER	18,470	48,365	0.614	0.575	0.048	political (left)

6 EXPERIMENTS

In this section, we study the PR and PPR fairness of a number of real networks, the effect of link recommendation algorithms in fairness and the edge characteristics that contribute to fairness the most. Our code and data are publicly available[1].

Graphs and their PR and PPR fairness: We use the following graphs:

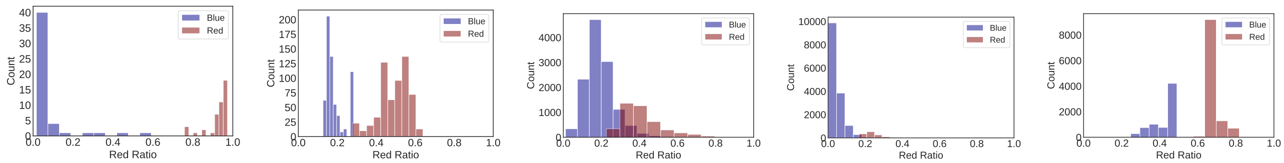
(1) BOOKS: A network of books about US politics where edges between books represented co-purchasing[2].
(2) BLOGS: A directed network of hyperlinks between weblogs on US politics [1].
(3) DBLP-GEN: An author collaboration network constructed from DBLP with a subset of data mining and database conferences from 2011 to 2020 with gender as the protected attribute. The value of gender is inferred using the Python gender guesser package[3].
(4) DBLP-PUB: The same network as DBLP-GENDER but with the protected group being the set of authors whose first publication appears in 2016, or later, i.e., the newcomers.
(5) TWITTER: A political retweet graph from [38].

The characteristics of the graphs are summarized in Table 1. We treat all graphs as directed. We define as *red*, the group whose PR ratio is smaller than its ratio in the overall population, that is the group to which the network is PR-unfair. This is the target group whose PR fairness we want to increase. For example, for the DBLP-GEN dataset, the red group is women. As seen, PR fairness varies among the graphs, some (e.g., BOOKS) are almost PR-fair (PR(R) ≈ $ratio(R)$), while others (e.g., BLOGS) are PR-unfair. We also report the *homophily* (h) of each graph that is the tendency of nodes to connect with nodes similar to them. For measuring homophily, we use $h = \frac{|cross\!-\!edges|/|E|}{2\,ratio(R)\,ratio(B)}$, where cross-edges are the edges connecting nodes belonging to different groups [12]. The closer h is to 0, the higher the homophily.

In Figure 1, we plot the distribution of the red PPR ratio for the red and blue nodes. In most datasets, blue nodes allocate most of their PPR to blue nodes, and red nodes to red nodes. In all datasets, there are nodes that are PPR-unfair, that is, the PPR they allocate to some group is smaller that the ratio of the group in the population. Most often, blue nodes are PPR-unfair to the red nodes, while red nodes are PPR-unfair to blue nodes. This is due to homophily.

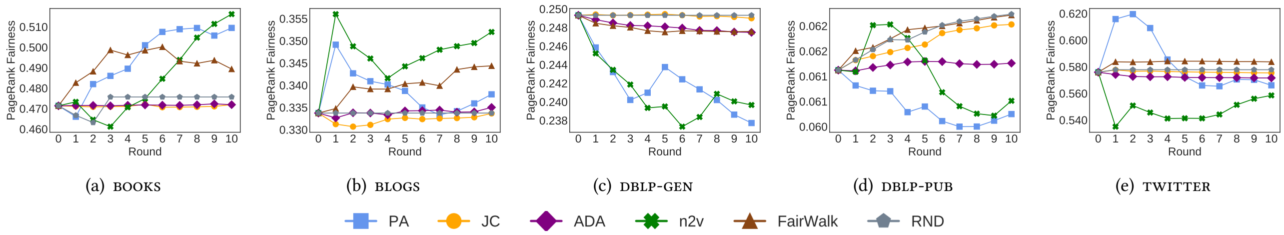
PR fairness and link recommendation algorithms. We now study the effect on PR fairness of link recommendation algorithms. To this end, we select 10% of the nodes randomly. Then, we add to

[1]https://github.com/ksemer/fairPRrec
[2]http://www-personal.umich.edu/ mejn/netdata/
[3]https://pypi.org/project/gender-guesser/



(a) BOOKS (b) BLOGS (c) DBLP-GEN (d) DBLP-PUB (e) TWITTER

Figure 1: Distribution of the red PPR ratio for the blue and red nodes.



(a) BOOKS (b) BLOGS (c) DBLP-GEN (d) DBLP-PUB (e) TWITTER

Figure 2: PR fairness (red PR ratio) for known link recommendation algorithms.

these nodes the 10 best edges as suggested by each of the recommendation algorithms. We add the edges in rounds, one edge at a time, and report the PR fairness towards the red group after each of the 10 rounds.

We start by studying a number of classic link recommendation algorithms. Specifically, we consider: (1) unsupervised recommendations based on scores, in particular, preferential attachment (PA), Jaccard Coefficient (JC), and Adamic-Adar (ADA) [29], (2) an embedding-based method node2vec (n2v) [21], and (3) FairWalk, an extension of node2vec that replaces random walks with fair random walks [36]. For computing the recommendations for the last two algorithms, we train a logistic regression classifier with the embeddings as features. For each node, in the case of unsupervised recommendation, we recommend the links with the highest score and for the last two algorithms, the links with the highest probabilities as estimated by the classifier. For comparison, we also consider random recommendations (RND).

As shown in Figure 2, overall, the difference between the red PR of the original network and the network after the recommendations is small. Recommendations based on local criteria (i.e., JC, ADA) do not affect the fairness of the network at all. Instead, we notice small fluctuations in the case of recommenders that favor central nodes (ie., PA, n2v) depending on the color that these central nodes have in each dataset.

Let us now turn to our algorithms, namely FREC and PREC. First, note that both recommend links to a node x based solely on the PR and PPR fairness gain respectively. That is, they ignore the probability $p_A(x, y)$ that x will accept the recommendation of edge (x, y). To address this, we introduce two variations (a) the *expected fair recommendation* (E_FREC) algorithm that selects edges based on the expected gain of the link, that is, $p_A(x, y)\, fgain(x, y)$, and (b) the *expected personalized fair recommendation* (E_PREC) algorithm that select edges based on the expected personalized gain, $p_A(x, y)\, pgain_x(x, y)$. Since the acceptance probability $p_A(x, y)$ of (x, y) is not known, we use as $p_A(x, y)$ the probability that the n2v classifier predicts for (x, y).

As shown in Figure 3, both the FREC and the E_FREC algorithms improve the PR-fairness. E_FREC achieves slightly smaller red PR values, since it also considers acceptance probabilities. In Table 2, we report the average acceptance probability of the edges recommended by each of the algorithms as these are estimated by node2vec. E_FREC increases fairness but also keeps the acceptance probabilities of the recommended edges high, achieving a good trade-off between fairness and accuracy.

Table 2: Average acceptance probability of recommended links (as estimated by n2v).

	BOOKS	BLOGS	DBLP-GEN	DBLP-PUB	TWITTER
RND	0.4297	0.3529	0.4131	0.4186	0.3655
n2v	0.5298	0.7827	0.8819	0.8213	0.7422
FairWalk	0.4360	0.3394	0.4238	0.4284	0.5268
FREC	0.4715	0.2996	0.4277	0.4285	0.5378
E_FREC	0.4930	0.6839	0.6746	0.5319	0.5688
PREC	0.4786	0.3048	0.4326	0.4463	0.6856
E_PREC	0.5047	0.7108	0.7205	0.5987	0.7982

Link recommendations for PPR fairness. We now study PPR fairness. For each node i, we increase the PPR that i allocates to the group S towards which i is PPR-unfair, that is, the group S for which $\overline{\mathbf{p}_i}(S) < ratio(S)$. In most cases, this is the opposite group of the group that node i belongs to (see, also Figure 1). By making the PPR of individual nodes fair, we expect that the percentage of $\mathbf{p}_i(S)$ allocated to each group S becomes less dependent on the color of i and comes closer to $ratio(S)$. In Figure 4, we plot the Wasserstein distance between the distribution of the percentage of $\mathbf{p}_i(R)$ (red PPR for short) for the blue nodes and for the red nodes. As shown, our algorithms PREC and E_PREC reduce the distance between these two distributions. Although E_PREC takes into account acceptance probabilities, it performs very similarly to PREC.

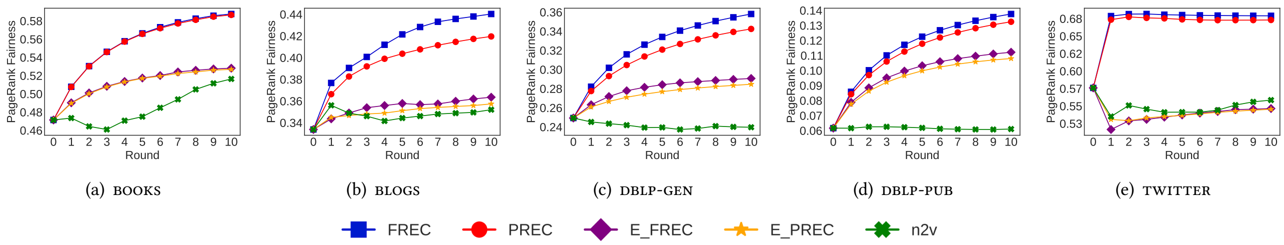


Figure 3: PR fairness (red PR ratio) of our algorithms.

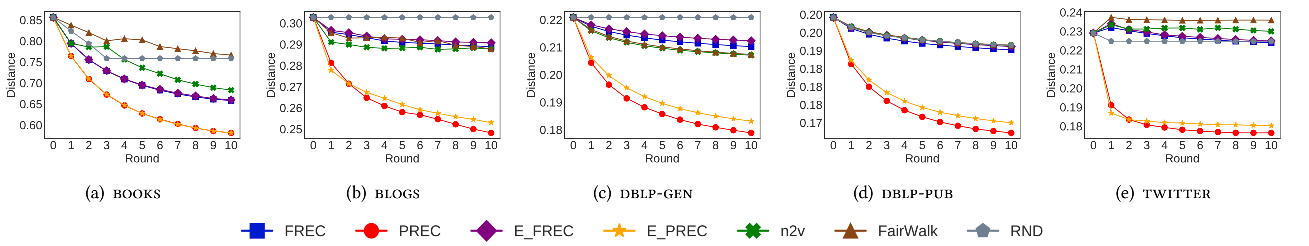


Figure 4: Wasserstein distances per round between the red PPR ratio of the blue nodes and the red PPR ratio of the red nodes.

In Figure 3, we also plot the PR-fairness achieved by the PREC and E_PREC algorithms. Despite the fact that PREC and E_PREC have a different objective, red PR is increased, when the red group is small (more evident for DBLP_GEN and DBLP_PUB). The reason is that due to homophily, the majority of the selected source nodes belong to the blue group and their PPR is unfair towards the red group and the algorithms increase the fairness to this group.

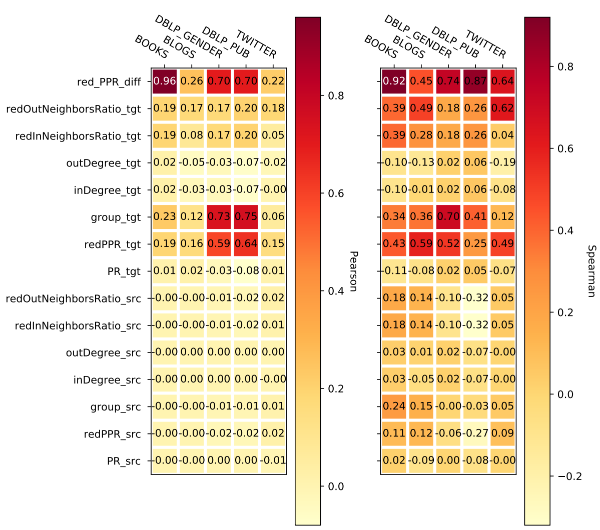


Figure 5: Correlation of edge characteristics and PR-fairness.

What makes an edge fairness-important? To see which characteristics of an edge are more relevant to PR-fairness, we compute the fairness value, $fvalue$, of all existing edges in the network and report in Figure 5 the correlation of this value with various characteristics of the source and target nodes of the edge.

A first observation is that the most important factor is the difference between the red PPR of the source and the red PPR of the target node (red_PPR_diff). This means that the most important edges in terms of fairness are the edges that connect nodes whose neighborhoods are of a "different color". Intuitively, this mean that the edges that connect heterogeneous (in term of color) parts of the graph are the most important ones for fairness.

A second observation is that in general the characteristics of the target node (suffix _tgt) of an edge have a stronger correlation with fairness than the characteristics of its source node (suffix _src). Among these characteristics, the most relevant are the group (i.e., color) and the red PPR of the target node. Edges pointing to nodes belonging to the red group are clearly important to fairness.

Finally, we see no important correlation for the PR and the degrees of both the source and the target nodes. Centrality of the edge endpoints is not in general strongly correlated with fairness, because only the central nodes for which the red_PPR_diff is positive increase fairness.

7 CONCLUSIONS

In this paper, we considered PageRank fairness. We derived analytical formulas to quantify the effect of existing and new edges on PageRank and personalized PageRank fairness. To improve fairness, we advocated an approach that aims at improving the fairness of the network itself. To achieve this, we proposed efficient liner-time link recommendation algorithms that suggest links that increase fairness. Our experimental results on real datasets have shown the effectiveness of our algorithms in creating fairer networks.

ACKNOWLEDGMENTS

Research work supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) Project No: HFRI-FM17-1873, GraphTempo.

REFERENCES

[1] Lada A. Adamic and Natalie S. Glance. 2005. The political blogosphere and the 2004 U.S. election: divided they blog. In *LinkKDD*, Jafar Adibi, Marko Grobelnik, Dunja Mladenic, and Patrick Pantel (Eds.). ACM, 36–43.

[2] Victor Amelkin and Ambuj K. Singh. 2019. Fighting Opinion Control in Social Networks via Link Recommendation. In *KDD*. 677–685.

[3] Chen Avin, Barbara Keller, Zvi Lotker, Claire Mathieu, David Peleg, and Yvonne-Anne Pignolet. 2015. Homophily and the Glass Ceiling Effect in Social Networks. In *ITCS*. 41–50.

[4] Konstantin Avrachenkov and Nelly Litvak. 2006. The effect of new links on Google PageRank. *Stochastic Models* 22, 2 (2006), 319–331.

[5] Avishek Joey Bose and William L. Hamilton. 2019. Compositional Fairness Constraints for Graph Embeddings. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. 715–724.

[6] S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems* 30, 1 (1998), 107 – 117.

[7] Balázs Csanád Csáji, Raphaël M. Jungers, and Vincent D. Blondel. 2014. PageRank optimization by edge selection. *Discret. Appl. Math.* 169 (2014), 73–87.

[8] Enyan Dai and Suhang Wang. 2021. Say No to the Discrimination: Learning Fair Graph Neural Networks with Limited Sensitive Attribute Information. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*. ACM, 680–688.

[9] Cristobald de Kerchove, Laure Ninove, and Paul Van Dooren. 2008. Maximizing PageRank via outlinks. *Linear Algebra Appl* 429, 5-6 (2008), 1274–1256.

[10] Peter G. Doyle and J. Laurie Snell. 1984. *Random Walks and Electric Networks*. Number Book 22 in Carus Mathematical Monographs. Mathematical Association of America, Washington, DC.

[11] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. 2012. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. 214–226.

[12] David A. Easley and Jon M. Kleinberg. 2010. *Networks, Crowds, and Markets - Reasoning About a Highly Connected World*. Cambridge University Press.

[13] Lisette Espín-Noboa, Claudia Wagner, Markus Strohmaier, and Fariba Karimi. 2021. Inequality and Inequity in Network-based Ranking and Recommendation Algorithms. *CoRR* abs/2110.00072 (2021).

[14] Francesco Fabbri, Francesco Bonchi, Ludovico Boratto, and Carlos Castillo. 2020. The Effect of Homophily on Disparate Visibility of Minorities in People Recommender Systems. In *ICWSM*. 165–175.

[15] Golnoosh Farnadi, Behrouz Babaki, and Michel Gendreau. 2020. A Unifying Framework for Fairness-Aware Influence Maximization. In *Companion of The 2020 Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*. 714–722.

[16] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. 2015. Certifying and Removing Disparate Impact. In *KDD*. 259–268.

[17] Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton, and Derek Roth. 2019. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*. 329–338.

[18] Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2017. Reducing Controversy by Connecting Opposing Views. In *WSDM*. 81–90.

[19] D. F. Gleich. 2015. PageRank Beyond the Web. *SIAM Rev.* 57, 3 (2015), 321–363.

[20] Charles M. Grinstead and J. Laurie Snell. 2003. *Introduction to Probability*. AMS. http://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/book.html

[21] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. 855–864.

[22] Shahrzad Haddadan, Cristina Menghini, Matteo Riondato, and Eli Upfal. 2021. RePBubLik: Reducing Polarized Bubble Radius with Link Insertions. In *WSDM*. 139–147.

[23] J. Kang, J. He, R. Maciejewski, and H. Tong. 2020. InFoRM: Individual Fairness on Graph Mining. In *KDD*. 379–389.

[24] Jian Kang and Hanghang Tong. 2021. Fair Graph Mining. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management*.

[25] Jian Kang, Meijia Wang, Nan Cao, Yinglong Xia, Wei Fan, and Hanghang Tong. 2018. AURORA: Auditing PageRank on Large Graphs. In *IEEE International Conference on Big Data, Big Data 2018, Seattle, WA, USA, December 10-13, 2018*. 713–722.

[26] Fariba Karimi, Mathieu Génois, Claudia Wagner, Philipp Singer, and Markus Strohmaier. 2018. Homophily influences ranking of minorities in social networks. *Nature Scientific Reports* 8 (2018).

[27] Matthäus Kleindessner, Samira Samadi, Pranjal Awasthi, and Jamie Morgenstern. 2019. Guarantees for Spectral Clustering with Fairness Constraints. In *Proceedings of the 36th International Conference on Machine Learning, ICML*. 3458–3467.

[28] Emmanouil Krasanakis, Symeon Papadopoulos, and Ioannis Kompatsiaris. 2020. Applying Fairness Constraints on Graph Node Ranks Under Personalization Bias. In *COMPLEX NETWORKS*, Vol. 944. Springer, 610–622.

[29] David Liben-Nowell and Jon M. Kleinberg. 2003. The link prediction problem for social networks. In *Proceedings of the 2003 ACM CIKM International Conference on Information and Knowledge Management, New Orleans, Louisiana, USA, November 2-8, 2003*. ACM, 556–559.

[30] Víctor Martínez, Fernando Berzal, and Juan Carlos Cubero Talavera. 2017. A Survey of Link Prediction in Complex Networks. *ACM Comput. Surv.* 49, 4 (2017), 69:1–69:33.

[31] Farzan Masrour, Tyler Wilson, Heng Yan, Pang-Ning Tan, and Abdol-Hossein Esfahanian. 2020. Bursting the Filter Bubble: Fairness-Aware Network Link Prediction. In *AAAI*.

[32] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.* 54, 6 (2021), 115:1–115:35.

[33] K. S. Miller. 1981. On the Inverse of the Sum of Matrices. *Mathematics Magazine* 54, 2 (1981), 67–72.

[34] Nikos Parotsidis, Evaggelia Pitoura, and Panayiotis Tsaparas. 2016. Centrality-Aware Link Recommendations. In *WSDM*. 503–512.

[35] Evaggelia Pitoura, Kostas Stefanidis, and Georgia Koutrika. 2021. Fairness in Rankings and Recommendations: An Overview. *The VLDB Journal* (2021).

[36] Tahleen A. Rahman, Bartlomiej Surma, Michael Backes, and Yang Zhang. 2019. Fairwalk: Towards Fair Graph Embedding. In *IJCAI*. 3289–3295.

[37] Bashir Rastegarpanah, Krishna P. Gummadi, and Mark Crovella. 2019. Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*. ACM, 231–239.

[38] Ryan A. Rossi and Nesreen K. Ahmed. 2015. The Network Data Repository with Interactive Graph Analytics and Visualization. In *AAAI*. http://networkrepository.com

[39] Ana-Andreea Stoica, Christopher J. Riederer, and Augustin Chaintreau. 2018. Algorithmic Glass Ceiling in Social Networks: The effects of social recommendations on network diversity. In *WebConf*. 923–932.

[40] Alan Tsang, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. 2019. Group-Fairness in Influence Maximization. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*. 5997–6005.

[41] Sotiris Tsioutsiouliklis, Evaggelia Pitoura, Panayiotis Tsaparas, Ilias Kleftakis, and Nikos Mamoulis. 2021. Fairness-Aware PageRank. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*. 3815–3826.

[42] Suresh Venkatasubramanian, Carlos Scheidegger, Sorelle A. Friedler, and Aaron Clauset. 2021. Fairness in Networks: Social Capital, Information Access, and Interventions. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. ACM, 4078–4079.

[43] Qinyong Wang, Hongzhi Yin, Hao Wang, Quoc Viet Hung Nguyen, Zi Huang, and Lizhen Cui. 2019. Enhancing Collaborative Filtering with Generative Augmentation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*. ACM, 548–556.

A PAGERANK AND PERSONALIZED PAGERANK FAIRNESS

We prove the following Lemma.

LEMMA A.1. *For a graph $G = (V, E)$, a group S, and the value $\phi = \frac{|S|}{|V|}$, it holds that if $\overline{\mathbf{p}_i}(S) \geq \phi$ for all $i \in V$, then $\mathbf{p}(S) \geq \phi$.*

PROOF. From the definition of $\overline{\mathbf{p}_i}(S)$, we have that:

$$\overline{\mathbf{p}_i}(S) = \begin{cases} \frac{\mathbf{p}_i(S)-\gamma}{1-\gamma}, & i \in S \\ \frac{\mathbf{p}_i(S)}{1-\gamma}, & i \notin S \end{cases}$$

For $\mathbf{p}(S)$ we have:

$$\begin{aligned} \mathbf{p}(S) &= \frac{1}{n}\sum_{i\in V}\mathbf{p}_i(S) \\ &= \frac{1}{n}\sum_{i\in S}\mathbf{p}_i(S) + \frac{1}{n}\sum_{i\notin S}\mathbf{p}_i(S) \\ &= \frac{1}{n}\sum_{i\in S}((1-\gamma)\overline{\mathbf{p}_i}(S)+\gamma) + \frac{1}{n}\sum_{i\notin S}(1-\gamma)\overline{\mathbf{p}_i}(S) \\ &= \frac{1}{n}(1-\gamma)\sum_{i\in V}\overline{\mathbf{p}_i}(S) + \gamma\frac{|S|}{n} \\ &\geq (1-\gamma)\phi + \gamma\phi \\ &= \phi \end{aligned}$$

The inequality follows from the fact that $\overline{\mathbf{p}_i}(S) \geq \phi$ for all $i \in V$, and $\phi = \frac{|S|}{n}$. □

Lemma A.1 says that if all nodes are PPR-fair to the group S, then PageRank is overall fair. The opposite is not necessarily true, PageRank may be fair, but there are individual nodes that are unfair.

B PROOF OF LEMMA 4.4

The proof of Lemma 4.4 relies on the absorbing random walks $\overline{X}$ we defined in Section 5.

We first prove the following Lemmas.

LEMMA B.1. *For every pair of nodes $i, j \in V$, $i \neq j$, it holds: $\mathbf{p}_j(i) < \mathbf{p}_i(i)$.*

PROOF. Let $f_{ji}^{(k)}$ be the probability to reach transient node i starting from transient node j for the first time at step k and $f_{ji}^* = \sum_{k=1}^{\infty} f_{ji}^{(k)}$. Let V_i be the number of visits to node i. It holds:

$$P[V_i = m \mid \overline{X}_0 = i] = {f_{ii}^*}^{m-1}(1-f_{ii}^*)$$

$$P[V_i = m \mid \overline{X}_0 = j] = \begin{cases} 1-f_{ji}^*, \ m=0 \\ f_{ji}^*{f_{ii}^*}^{m-1}(1-f_{ii}^*) \end{cases}$$

Thus V_i follows a geometric distribution with success probability $(1-f_{ii}^*)$ and so:

$$E[V_i \mid \overline{X}_0 = i] = \frac{1}{1-f_{ii}^*}$$

$$E[V_i \mid \overline{X}_0 = j] = f_{ji}^*\, E[V_i \mid \overline{X}_0 = i]$$

Since there is a nonzero probability to reach i, i.e., $f_{ji}^* \neq 0$:

$$E[V_i \mid \overline{X}_0 = j] < E[V_i \mid \overline{X}_0 = i]$$

From $E[V_i \mid \overline{X}_0 = j] = \overline{\mathbf{F}}_{ji} = \frac{\mathbf{Q}_{ji}}{\gamma}$ and $E[V_i \mid \overline{X}_0 = i] = \overline{\mathbf{F}}_{ii} = \frac{\mathbf{Q}_{ii}}{\gamma}$, we get $\mathbf{p}_j(i) < \mathbf{p}_i(i)$. □

LEMMA B.2. *For the personalized PageRank that node i gives to itself, it holds:*

$$\mathbf{p}_i(i) = \gamma + (1-\gamma)\frac{1}{d_i}\sum_{w\in N_i}\mathbf{p}_w(i)$$

PROOF. The proof follows directly from the fact that

$$\mathbf{p}_i^T = \gamma\mathbf{e}_i + (1-\gamma)\mathbf{p}_i^T\mathbf{P}$$

□

For the proof of Lemma 4.4 it suffices to show that the denominator of $\Lambda((x, y), S)$ in Theorem 4.1 is always positive. From Lemma B.2:

$$d_x \ + \ 1 \ - \ \frac{(1-\gamma)}{\gamma}\left(\mathbf{p}_y(x) - \frac{1}{d_y}\sum_{w\in N_x}\mathbf{p}_w(y)\right) =$$

$$d_x - \frac{1}{\gamma}\left((1-\gamma)\mathbf{p}_y(x) - \mathbf{p}_x(x)\right)$$

This quantity is always positive since from Lemma B.1, $\mathbf{p}_y(x) < \mathbf{p}_x(x)$ and $1-\gamma < 1$.

C PROOF OF THEOREM 4.5

The proof follows closely that of Theorem 4.1. Similar to before we express the addition of the edges in E_x as a perturbation of the transition probability matrix $\mathbf{P}$ of PageRank with a rank-1 matrix $\mathbf{D}$:

$$\mathbf{P'} = \mathbf{P} + \mathbf{D}, \quad \mathbf{D}_i = \begin{cases} 0, \ i \neq x \\ -\frac{k}{d_x+k}\mathbf{P}_x + \frac{1}{d_x+k}\mathbf{e}_{E_x}^T, \ i = x \end{cases}$$

where $\mathbf{e}_{E_x}$ is the vector with 1 at the positions of the added edges, and zero everywhere else.

The updated matrix $\mathbf{Q'}$ can be computed using Equation 5. With mathematical manipulations, we get:

$$\mathbf{DQ}_{ij} = \begin{cases} 0, \ i \neq x \\ \frac{k}{d_x+k}\left(\frac{1}{k}\sum_{y\in V_y}\mathbf{Q}_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wj}\right), \quad i = x \end{cases}$$

$$\mathbf{QDQ}_{ij} = \frac{k}{d_x+k}\mathbf{Q}_{ix}\left(\frac{1}{k}\sum_{y\in V_y} Qmatrix_{yj} - \frac{1}{d_x}\sum_{w\in N_x}\mathbf{Q}_{wj}\right)$$

Substituting in (5), and summing over $j \in S$, we obtain Theorem 4.5(2). Summing over $i \in V$ we obtain Theorem 4.5(1).

D PROOF OF THEOREM 4.6

PROOF. For the case that $d_x > 1$, the proof proceeds similarly with the proof of Theorem 4.1. We first write the transition matrix $\mathbf{P}$ of G' as a sum of the transition matrix $\mathbf{P}$ of G and a rank one, perturbation matrix $\mathbf{D}$. This is:

$$\mathbf{P'} = \mathbf{P} + \mathbf{D}, \quad \mathbf{D}_i = \begin{cases} 0, \ i \neq x \\ \frac{1}{d_x-1}\mathbf{P}_x - \frac{1}{d_x-1}\mathbf{e}_y, \ i = x \end{cases}$$

As in Theorem 4.1, we get Equation 5 by using the fundamental theorem for the inverse of the sum of matrices and the formula for Q from Lemma 3.1.

Algorithm 2 Greedy Algorithm

Require: Graph $G(V, E)$, source node $x \in V$, under-represented group R, value k

1: Compute PageRank vector $\mathbf{p}$ on G
2: **for** each $v \in V$ **do**
3: Compute $p_v(R)$ on graph G
4: **end for**
5: **for** each $v \in V$ **do**
6: Compute $p_v(x)$ on graph G
7: **end for**
8: $S = \emptyset$
9: **for** $i = 1 \ldots k$: **do**
10: **for** each $v \in V$: $(x, v) \notin E \cup S$ **do**
11: Compute $fdelta(S, (x, v)) = fgain(S \cup \{(x, v)\}) - fgain(S)$
12: **end for**
13: $(x, y) = \arg\max_{(x,v)} fdelta(S, (x, v))$
14: $S = S \cup \{(x, y)\}$
15: **end for**
16: **return** S

With mathematical manipulations, we get:

$$\mathbf{DQ}_{ij} = \begin{cases} 0, \; i \neq x \\ \frac{1}{d_x - 1} \left(\frac{1}{d_x} \sum_{w \in N_x} \mathbf{Q}_{wj} - \mathbf{Q}_{yj} \right), \quad i = x \end{cases}$$

$$\mathbf{QDQ}_{ij} = \frac{1}{d_x - 1} \mathbf{Q}_{ix} \left(\frac{1}{d_x} \sum_{w \in N_x} \mathbf{Q}_{wj} - \mathbf{Q}_{yj} \right)$$

and substituting in Equation 5:

$$\mathbf{Q}'_{ij} = \mathbf{Q}_{ij} + \mathbf{Q}_{ix} \frac{\frac{(1-\gamma)}{\gamma} \left(\frac{1}{d_x} \sum_{w \in N_x} \mathbf{Q}_{wj} - \mathbf{Q}_{yj} \right)}{d_x - 1 - \frac{(1-\gamma)}{\gamma} \left(\frac{1}{d_x} \sum_{w \in N_x} \mathbf{Q}_{wx} - \mathbf{Q}_{yx} \right)}$$

Now, using Lemma 3.1 as in Theorem 4.1, we get the formula in Theorem 4.6.

Special care is required in defining the perturbation matrix $\mathbf{D}$ for the case that $d_x = 1$. In this case, when removing the single edge out of node x, the entry $\mathbf{P}_x$ in the transition matrix becomes the uniform matrix. Therefore, we have:

$$\mathbf{D}_i = \begin{cases} 0, \; i \neq x \\ \mathbf{u} - \mathbf{e}_y, \; i = x \end{cases}$$

where $\mathbf{u}$ is the uniform vector with $1/|V|$ in all entries.

With mathematical manipulations, we get:

$$\mathbf{DQ}_{ij} = \begin{cases} 0, \; i \neq x \\ \frac{1}{|V|} \sum_{w \in V} \mathbf{Q}_{wj} - \mathbf{Q}_{yj}, \quad i = x \end{cases}$$

$$\mathbf{QDQ}_{ij} = \mathbf{Q}_{ix} \left(\frac{1}{|V|} \sum_{w \in V} \mathbf{Q}_{wj} - \mathbf{Q}_{yj} \right)$$

and substituting in Equation 5:

$$\mathbf{Q}'_{ij} = \mathbf{Q}_{ij} + \mathbf{Q}_{ix} \frac{\frac{1-\gamma}{\gamma} \left(\frac{1}{|V|} \sum_{w \in V} \mathbf{Q}_{wj} - \mathbf{Q}_{yj} \right)}{1 - \frac{1-\gamma}{\gamma} \left(\frac{1}{|V|} \sum_{w \in V} \mathbf{Q}_{wx} - \mathbf{Q}_{yx} \right)}$$

Now, using Lemma 3.1 as in Theorem 4.1, we get the formula in Theorem 4.6. □

E OUTLINE OF THE GREEDY ALGORITHM

The outline of the Greedy algorithm described in Section 5.2 is shown in Algorithm 2. In lines 2-7 we compute the quantities $\mathbf{p}_v(R)$ and $\mathbf{p}_v(x)$ using two PageRank-like computations as described in BFE. In lines 9-15, we construct the set of edges S. Line 11 can be computed in constant time as explained. Therefore, the complexity of the algorithm is $O(k|V| + |E|)$.