Optimizing Rankings for Recommendation in Matching Markets

Yi Su
Cornell University
Ithaca, NY, USA
ys756@cornell.edu

Magd Bayoumi
Cornell University
Ithaca, NY, USA
mb2363@cornell.edu

Thorsten Joachims
Cornell University
Ithaca, NY, USA
tj@cs.cornell.edu

ABSTRACT

Based on the success of recommender systems in e-commerce and entertainment, there is growing interest in their use in matching markets like job search. While this holds potential for improving market fluidity and fairness, we show in this paper that naively applying existing recommender systems to matching markets is sub-optimal. Considering the standard process where candidates apply and then get evaluated by employers, we present a new recommendation framework to model this interaction mechanism and propose efficient algorithms for computing personalized rankings in this setting. We show that the optimal rankings need to not only account for the potentially divergent preferences of candidates and employers, but they also need to account for capacity constraints. This makes conventional ranking systems that merely rank by some local score (e.g., one-sided or reciprocal relevance) highly sub-optimal — not only for an individual user, but also for societal goals (e.g., low unemployment). To address this shortcoming, we propose the first method for jointly optimizing the rankings for all candidates in the market to explicitly maximize social welfare. In addition to the theoretical derivation, we evaluate the method both on simulated environments and on data from a real-world networking-recommendation system that we built and fielded at a large computer science conference.

CCS CONCEPTS

• **Information systems → Probabilistic retrieval models**; • **Computing methodologies → Ranking**.

KEYWORDS

Multi-sided Markets, Ranking, Social Welfare

ACM Reference Format:
Yi Su, Magd Bayoumi, and Thorsten Joachims. 2022. Optimizing Rankings for Recommendation in Matching Markets. 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/10.1145/3485447.3511961

1 INTRODUCTION

Most search and recommender systems rely on rankings as the prevalent way of presenting results to the users. By ranking the most promising items first, they aim to focus the user's attention on a tractably small consideration set especially when the number of items is large. For conventional applications of ranking systems in online retail or media streaming, it has long been understood that ranking items based on their probability of relevance (alternatively, purchase or click) to the user provides maximal utility under standard assumptions. Sorting by the probability of relevance is called the Probability Ranking Principle (PRP) [29], and it implies that the optimal ranking for any user depends only on their preferences.

The PRP is no longer optimal, however, for a growing range of new online platforms that mediate *matching* markets like job search, college admission, dating, or social recommendation [4, 15, 21, 38]. The following two key properties of matching markets make the ranking problem substantially more complex than in conventional two-sided markets. First, a successful match requires that both sides of a match agree on mutual relevance [14, 26]. Second, both sides have constraints on how many options they can evaluate. Under these conditions, the optimal ranking for any one participant depends on the rankings and relevances of the other participants in the market, which can make PRP ranking highly suboptimal.

To illustrate the source of these dependencies, consider a job matching problem where many of the job candidates crowd to a small set of employers with high name-recognition. The popular employers will receive many applications — more than they can carefully evaluate and many without the desired qualifications. At the same time, less well-known employers may not get discovered by relevant candidates. The result is that many candidates never hear back from the popular employers and many employers are unable to fill their positions. This is not only bad for individual employers and candidates, but also a poor outcome in terms of the social welfare that the system provides (e.g. minimizing unemployment and unfilled jobs). To remedy this problem, a recommender system for job matching should consider preferences and qualifications from both the candidates (*e.g.,* west-coast location, python) and the employers (*e.g.,* 5 years experience, python) and help discover mutually beneficial matches that avoid crowding.

The main contributions of this paper are fourfold: (1) First, the paper is the first to formalize the problem of *ranking in two-sided matching markets* under the apply/accept interaction protocol that is used in many real-world settings. (2) Second, we present the first recommender system that is able to jointly optimize personalized rankings for all candidates to maximize social welfare in two-sided matching markets. (3) Third, we discuss additional strategic behavior (e.g., adoption and retention) and fairness issues for rankings under this framework, which opens an important area for future work. (4) Finally, we built a real-world networking-recommendation system that we fielded at a large computer science conference. We empirically validate our method on this real-world dataset, which we provide as a benchmark to enable future research.

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3511961

1.1 Overview of Results

Before going into the formal and mathematical details of our approach, we provide an overview of the model and the main results.

Modeling matching markets using the apply/accept protocol: We consider the apply/accept interaction protocol that is dominantly used in matching markets like job search and college admission. Focusing on the example of job search for clarity, job candidates on one side of the market act by sending their applications to employers they are interested in. The employers on the other side of the market respond after evaluating the applications they receive, and a positive response leads to the mutually desirable outcome of a match. A key component of our framework is that both the candidates and the employers have limited attention, and thus have to act under partial information. In particular, the candidates cannot evaluate all job postings, and thus require a personalized ranking that focuses their attention on the most promising job postings. Similarly, employers want a ranking of applications they receive to triage the review process and better allocate reviewing resources. This leads to a two-sided ranking problem which we formalize, and where we identify the total number of mutually relevant matches as the social-welfare objective (i.e. minimizing unemployment and unfilled jobs). Note that this two-sided ranking problem is fundamentally new compared to prior work [4, 14, 15, 21, 26, 38].

Optimizing for attention and discovery: The goal of our framework is different from optimizing matchings under known preferences as in traditional stable matching procedures from economics [10, 16]. Instead, we support the **preference formation** on both sides of the market when the number of options is large. We thus aim to maximize social-welfare by focusing the attention budget of both sides on the most promising options for manual evaluation, which is analogous to the attention-focusing goals of conventional recommender systems. In this way, our approach supports each participant in discovering the most promising options to manually evaluate, optimizing how the participants spend their scarce resource of attention among an often overwhelming set of options. This leaves all participants with autonomy over their decisions, which makes it different from conventional matching procedures [10, 16], where all participants have to submit to a centralized matching procedure [31] that determines the matching outcome.

Capturing mutual relevance and crowding: Under the apply/accept protocol, rankings for the candidates become dependent given the limited attention budget of the employers. We are the first to model this dependency when both the employer and the candidates receive their recommendations as rankings. In particular, crowding of candidates to well-known employers influences whether an applicant will be noticed by an employer and receive a reply – and thus the priority a crowded employer should have for evaluation by other candidates. We, therefore, model the position each candidate would reach in an employer ranking, which depends on the application behavior of all other candidates. Our model is the first to capture social-welfare of a ranking system under the dependencies introduced by crowding and mutual relevance.

Method for finding rankings that optimize social-welfare: Our key algorithmic contribution is the first method for computing rankings that optimize social welfare in two-sized matching markets. This objective requires us to jointly compute the rankings for all candidates, as the utility for each candidate also depends on how others act in the market. This is fundamentally different from traditional ranking algorithms, where each user's ranking is optimized independently. To derive a tractable algorithm, we use three key technical steps. First, we show how to transform the discrete and exponentially-sized optimization problem over rankings into a stochastic optimization problem in the polynomially-sized space of marginal rank distributions. Second, we derive a lower bound on the social-welfare objective and show how this objective can be computed and optimized efficiently. Finally, we exploit the Birkhoff-von Neumann decomposition to recover rankings from the compact representation we used during optimization.

Providing potential for incorporating strategic behaviour: Beyond developing algorithms for maximizing the overall utility of the market, we also initiate a discussion of fairness and strategic behaviors (such as user adoption and retention) for ranking in matching markets. These issues are more complex than in traditional ranking systems, as the behaviour of individuals affects each other. However, our optimization-based framework provides a promising potential for incorporating various constraints, such as statistical-parity and merit-based exposure constraints. All of these open an important area for future work about this new type of ranking systems in matching markets.

1.2 Related Work

As *multi-sided market platforms* have emerged as a popular business model, search and recommender systems have taken a key role in mediating their interactions. These systems need to balance the interests of various stakeholders, such as the users, the suppliers, and the platform itself. There is a large body of recent work on how to specify each stakeholder's objective [17], their interplay [18, 41], and the design of efficient joint optimization objectives for recommendation in multi-sided marketplaces [19, 30, 36, 45]. Important objectives include diversity [7, 27], novelty [28, 39] and fairness [5, 6, 33, 44]. Most of the work relies on techniques from multi-objective optimization, which include finding the Pareto front [23], using aggregation functions that reduce to a single objective [13, 19, 45], or including some objectives as constraints [30, 33, 34]. Unlike most of these works, we directly model the interaction process in two-sided matching markets and aim to maximize the overall social welfare of the market as the key objective.

Our problem setting is a form of *reciprocal recommendation* [14, 26], which considers matching problems where both sides have preferences like online dating [38, 42], job seeking [2, 20, 43], and social media [46]. Hence, most reciprocal recommendation systems rank by a reciprocal score that combines the preferences from both sides, like the harmonic mean [25], sum of similarities [2], product operator [37], and community-level matching [3]. Among these works, [38] is the closest to ours. They propose a two-sided matching framework for dating recommendation that maximizes the total number of reciprocated messages, while avoiding overburdening each user by keeping the expected number of messages received/sent bounded. A key simplifying assumption in [38] is that the system provides a *predetermined* number of recommendations to each user, which is typically difficult to pick efficiently in practice, and then all these recommendations will be exhaustively evaluated by the user. Similar assumptions are also made by recent work on assortment planning [4] for matching problems. We consider the more realistic and strictly more general case where each user is shown a *ranking*, and there is uncertainty in how far each user goes down. Also, we model the widely used apply/accept protocol, which was not supported in existing methods.

Our work also builds upon the economic and social science literature on *matching markets* [10, 16]. Here, the core problem lies in the design of matching procedures. In particular, stable-matching algorithms [10] take a ranked preference list from both sides of the market as input and produce stable one-to-one or many-to-one matchings, while maximum weight matching [9] considers weighted preference graphs. In contrast, our work helps the users *form* their preferences despite an intractably large set of possible options. In particular, we do not force the final matching step and leave all participants with autonomy over their actions and eventual decisions. This makes it fundamentally different from conventional matching procedures.

2 A FRAMEWORK FOR RANKING IN MATCHING MARKETS

We begin by introducing our new framework for rankings in matching markets. For simplicity and concreteness, we use a *job recommendation platform* as a running example. However, the framework is general and can be adapted to other domains where (1) the success of a matching relies on relevance from both sides and (2) both sides have limited attention.

We consider the following *sequential* and *asymmetric* interaction protocol. First, *job candidates* (proactive side) browse through their ranked lists of jobs and apply to the jobs that they find relevant. After this, *employers* (reactive side) browse through rankings of their applicants and respond to the applicants they find relevant (e.g., invite to interview). We call this a match, and the goal of the ranking system is to design rankings for each candidate to maximize the total number of expected matches in the market.

We now formalize this interaction model, and then present algorithms for jointly optimizing the rankings for all candidates in Section 3. The toy example in Figure 1 serves as an intuitive guide to the model we develop in the following. We denote the set of candidates in the market as C, and the set of employers as $\mathcal{J}$ with $|C| < \infty$ and $|\mathcal{J}| < \infty$. For simplicity, we use binary relevance $r(c, j) \in \{0, 1\}$ to denote how employer j is relevant to candidate c (*i.e., c* wants to apply to j). Similarly, $\tilde{r}(j, c) \in \{0, 1\}$ denotes the relevance of candidate c to employer j (*i.e., j* wants to interview c). However, the ranking system does not know the precise relevances, but it merely has access to relevance probabilities

$$f_c(j) := \mathbb{P}(r(c, j) = 1) \quad \text{and} \quad g_j(c) := \mathbb{P}(\tilde{r}(j, c) = 1).$$

A large body of literature exists on how to estimate probabilities of relevance, and for the purposes of this paper we assume that accurate and unbiased estimates exist.

2.1 Candidates Act (Proactive)

Like in many real-world markets, candidates act first in our model. Our recommendation system is characterized by a *contextualized stochastic* ranking policy $\pi : C \longrightarrow \Delta_{\Sigma_{|\mathcal{J}|}}$, which maps each candidate to a probability distribution over rankings of the employers. The space of all possible rankings is denoted by $\Sigma_{|\mathcal{J}|}$. For each candidate c, the system samples a deterministic ranking $\sigma(c)$ according to $\pi(\cdot|c)$, *i.e.,* $\sigma(c) \sim \pi(\cdot|c)$ and presents it to the candidate c.

After each candidate c receives a ranking of employers $\sigma(c)$, they act independently by going down the ranking and applying to an employer if it is relevant. We use the position-based model (PBM) [12] to model this application process. Given a ranking $\sigma(c)$, the PBM models the probability of candidate c applying to job j as the product of the relevance probability $f_c(j)$ and the examination probability $\mathbb{P}(e(j) = 1|\sigma(c))$:

$$\mathbb{P}(c \text{ applies to } j|\sigma(c)) = f_c(j) \cdot \mathbb{P}(e(j) = 1|\sigma(c)) \tag{1}$$

The examination probability models how likely a candidate will discover a job that is placed in a particular rank. In the PBM, this examination probability depends only on the rank $rank(j|\sigma(c))$ of employer j under ranking $\sigma(c)$. Therefore, we can write

$$\mathbb{P}(e(j) = 1|\sigma(c)) = v(rank(j|\sigma(c))), \tag{2}$$

where v is an application-dependent function. Common choices include $v(x) = 1/x$ [12] and $v(x) = \frac{1}{\log(1+x)}$ [11], or an application-dependent v can be estimated directly [1]. Combining this, the probability $\mathbb{P}^{\pi}_{c,j}$ of candidate c applying to employer j under ranking policy π can be expressed as:

$$\mathbb{P}^{\pi}_{c,j} := \mathbb{P}(c \text{ applies to } j|\pi) = \sum_{\sigma(c) \in \Sigma_{|\mathcal{J}|}} \pi(\sigma(c)|c) f_c(j) v\big(rank(j|\sigma(c))\big). \tag{3}$$

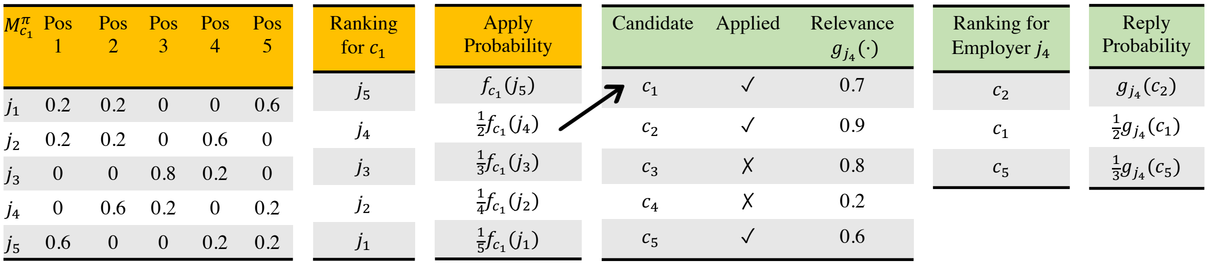
This expression may appear intractable at first glance, since it involves a sum over the exponential number of possible rankings. However, note that the relevance probability $f_c(j)$ does not depend on the rank and that the examination probability $v\big(rank(j|\sigma(c))\big)$ only depends on rank, such that $\mathbb{P}^{\pi}_{c,j}$ can be expressed equivalently in terms of the marginal probabilities $\mathbb{P}(rank(j|\sigma(c))) = k|\pi)$ of employer j being placed at rank k under policy π. Therefore, $\mathbb{P}^{\pi}_{c,j}$ can be re-written as:

$$\mathbb{P}^{\pi}_{c,j} = f_c(j) \Big(\sum_{k=1}^{|\mathcal{J}|} \mathbb{P}(rank(j|\sigma(c)) = k|\pi) v(k) \Big) = f_c(j) \Big(\sum_{k=1}^{|\mathcal{J}|} M^{\pi}_c(j,k) v(k) \Big)$$

M^{π}_c denotes the doubly stochastic matrix for candidate c where the (j, k)-th position equals $\mathbb{P}(rank(j|\sigma(c))) = k|\pi)$. This enables us to use only $|\mathcal{J}|^2$ parameters when representing ranking policy $\pi(\cdot|c)$ for candidate c via M^{π}_c, since all stochastic ranking policies with the same M^{π}_c are equivalent. Meanwhile, we use $\mathbf{M}^{\pi}_C$ as the concatenation of M^{π}_c $\forall c \in C$ to denote the set of doubly stochastic matrices for all candidates. In the following, this will allow us to optimize in the space of doubly stochastic matrices instead of in the exponentially sized space of rankings. Once an optimal M^{π}_c is found, we use the Birkhoff-von Neumann (BvN) decomposition to efficiently find a stochastic ranking policy that corresponds to M^{π}_c in polynomial time, and then sample a single ranking to present to the candidate [33].

To further compact the notation, we write the examination probabilities[1] as a vector $\mathbf{v} \in \mathbb{R}^{|\mathcal{J}|}_{+}$ with $\mathbf{v}[k] = v(k)$ for $k \in$

[1] Here, we assume user-homogeneous examination. It could be easily extended into user-heterogenous case.



$M_{c_1}^{\pi}$	Pos 1	Pos 2	Pos 3	Pos 4	Pos 5
j_1	0.2	0.2	0	0	0.6
j_2	0.2	0.2	0	0.6	0
j_3	0	0	0.8	0.2	0
j_4	0	0.6	0.2	0	0.2
j_5	0.6	0	0	0.2	0.2

Ranking for c_1	Apply Probability
j_5	$f_{c_1}(j_5)$
j_4	$\frac{1}{2}f_{c_1}(j_4)$
j_3	$\frac{1}{3}f_{c_1}(j_3)$
j_2	$\frac{1}{4}f_{c_1}(j_2)$
j_1	$\frac{1}{5}f_{c_1}(j_1)$

Candidate	Applied	Relevance $g_{j_4}(\cdot)$
c_1	✓	0.7
c_2	✓	0.9
c_3	✗	0.8
c_4	✗	0.2
c_5	✓	0.6

Ranking for Employer j_4	Reply Probability
c_2	$g_{j_4}(c_2)$
c_1	$\frac{1}{2}g_{j_4}(c_1)$
c_5	$\frac{1}{3}g_{j_4}(c_5)$

Figure 1: A two-sided matching market with 5 candidates, 5 employers, and examination function $v(x) = 1/x$. The yellow tables show the interaction process for a particular candidate c_1, while the green tables are for a particular employer j_4 (other candidates and employers follow an analogous process). From left to right, the system first computes a personalized stochastic ranking $M_{c_1}^{\pi}$ (the 5×5 matrix) for c_1. Then it samples a ranking $(j_5, j_4, j_3, j_2, j_1)$, presents it to c_1, and c_1 applies to the employers according to the position-based model. For the employer side, one realization of the process shows $\{c_1, c_2, c_5\}$ applying to j_4. The system thus presents ranking (c_2, c_1, c_5) to j_4 sorted by $g_{j_4}(\cdot)$. Finally, j_4 replies back also following a position-based model.

$\{1, 2, 3, \cdots, |\mathcal{J}|\}$, and we use $\mathbf{e}_j$ as the standard basis vector in $|\mathcal{J}|$-dimensional vector space, with a 1 in the j-th position and 0's elsewhere. Now the probability of candidate c applying to employer j can be written as: $\mathbb{P}_{c,j}^{\pi} = f_c(j)\mathbf{e}_j^T M_c^{\pi}\mathbf{v}$.

2.2 Employers Act (Reactive)

After collecting all the applications, it is the employers' turn to reply back. To be specific, the ranking system will give each employer j a ranked list of candidates *who applied to them*, sorted by their employer-specific probability of relevance $g_j(\cdot)$. We denote the set of candidates who applied to employer j as $C_j^{\pi} \subseteq C$, and it is important to point out that C_j^{π} is stochastic and the randomness comes from the candidates' application outcome under policy π. Then each employer responds back to the candidates $c \in C_j^{\pi}$ following a similar PBM model as the one we introduced for the candidates[2]. In particular, the probability that employer j replies back to candidate c (e.g., invites to interview) depends on the relevance probability $g_j(c)$ and the examination probability. Note that the examination probability for any candidate c who applied to j depends on who else is in C_j^{π}, inducing a distribution over ranks. Let $\mathbf{rk}_j^{\pi}(c)$ denote the rank of candidate c in a particular C_j^{π} given c applied to j, ranked by $g_j(c)$. We can then write the probability of a reply as follows:

$$\mathbb{P}_{j,c}^{\pi} := \mathbb{P}(j \text{ replies to } c | c \text{ applied to } j, \pi) = g_j(c)\mathbb{E}\left[v\left(\mathbf{rk}_j^{\pi}(c)\right)\right] \tag{4}$$

The expectation is over the randomness in C_j^{π}, which is induced by the application process on the side of the candidates.

2.3 Utility and Social Welfare

We now formulate the objective of the ranking system, which is based on the notion of successful matches. A match for a candidate/employer pair (c, j) is successful, if candidate c manages to discover a relevant j and thus applies, and employer j also finds c relevant and manages to discover this candidate among the applicants. We use $Y_{c,j}^{\pi} \in \{0, 1\}$ to denote whether a match is successful. We define the utility $\mathbf{U}_c(\pi)$ that the ranking system provides to candidate c when using ranking policy π as the expected number of matches (e.g., interviews) that c receives.

$$\begin{aligned}\mathbf{U}_c(\pi) &= \sum_{j\in\mathcal{J}} \mathbb{P}(Y_{c,j}^{\pi} = 1) = \sum_{j\in\mathcal{J}} \mathbb{P}_{c,j}^{\pi}\mathbb{P}_{j,c}^{\pi} \\ &= \sum_{j\in\mathcal{J}} \left(f_c(j)g_j(c)\mathbb{E}\left[v\left(\mathbf{rk}_j^{\pi}(c)\right)\right]\right)\mathbf{e}_j^T M_c^{\pi}\mathbf{v}\end{aligned} \tag{5}$$

After rearranging the terms, we can see that the utility of ranking policy π for candidate c is analogous to traditional ranking measures like DCG [11]. The key difference is that the utility for each ranked item not only depends on the candidates' own relevance, but also on those of the employers and potentially all other candidates, since they affect the position of this candidate in the employers' rankings. One can also define a similar utility $\mathbf{U}_j(\pi)$ for each employer.

Given this *interdependent* nature of the individual utilities, and the societal role that many of these matching markets play, we focus on *social welfare* as the overall objective of the ranking system. We select the most straightforward definition of social welfare $\mathbf{SW}(\pi)$, which is the expected number of matches in the market that ranking policy π produces. Note that this is equivalent to maximizing the sum of candidate utilities, or equivalently the sum of employer utilities.

$$\mathbf{SW}(\pi) = \sum_{c\in C} \mathbf{U}_c(\pi) = \sum_{j\in\mathcal{J}} \mathbf{U}_j(\pi) \tag{6}$$

Maximizing the total number of matches is arguably a sensible proxy for societal goals (e.g., minimize unemployment), although one could further refine this objective (e.g. diminishing returns for individuals [22]) for other objectives. In particular, an important additional consideration is ensuring the fairness of such a ranking system, which we futher discuss in Section 4.3.

In summary, we define the problem of *Social-Welfare Maximization of Rankings for Two-Sided Matching Markets* that we consider in this paper as follows.

Definition 1. *Given a two-sided matching market with a proactive side C and a reactive side $\mathcal{J}$, along with two-sided relevance*

[2] In general, the examination function $v(\cdot)$ could be different for candidates and employers. Here we keep it the same for simplicity of notation.

probability $f_c(j), g_j(c)\ \forall c \in C, j \in \mathcal{J}$, the goal is to design a stochastic ranking policy π that maximizes the expected number of matches in the market.

While an abstraction of the real-world, our model captures the key interactions and trade-off's faced in many modern two-sided matching markets, where (1) the number of available options is often very large, (2) both sides of the market do not have the resources to evaluate all options exhaustively, and (3) a successful transaction depends on a two-sided relevance match. In reality, the interaction for two sides of the market might be a highly dynamic process. We argue that our framework could be adapted to this by re-optimizing the rankings repeatedly conditioning on any new information available in the market.

It is worth pointing out that our ranking problem is fundamentally different from stable matching problems considered in the economics literature [10, 16]. First, our ranking problem aims at assisting users in forming their preferences over the options by focusing the user's attention, in contrast to assuming that users can readily state their preferences for the matching procedure. However, it still leaves open the question whether stable matching procedures could be used to solve our optimization problem.

To address this question, consider the following simplified *top-1 ranking problem* where all participants only examine position one of their ranking (i.e., $v(1) = 1$ and $v(k) = 0$ for all $k \geq 2$ and for both sides). Furthermore, we assume that the relevance probabilities $f_c(j)$ and $g_j(c)$ imply a preference order for each candidate and employer. The following proposition clarifies that even in this simplified setting stable matching and social-welfare maximization are not equivalent. propositioncomparemm A matching that is stable is not necessarily a social-welfare optimal top-1 ranking, and top-1 rankings that are not stable matchings can have better social welfare than stable matchings.

c_1	j_1	j_3	j_2
c_2	j_2	j_1	j_3
c_3	j_1	j_2	j_3
$f_c(j)$	1	0.9	0.1

j_1	c_1	c_3	c_2
j_2	c_2	c_1	c_3
j_3	c_1	c_2	c_3
$g_j(c)$	1	0.9	0.1

Proof. We prove the claim by considering the two-sided market shown above. There are three job candidates $\{c_1, c_2, c_3\}$ and three employers $\{j_1, j_2, j_3\}$. The candidates' relevance table is shown at left, where each row is a ranked list of employers sorted by the candidate-side relevance probability $f_c(j)$. Similarly, the table on the right is the ranked list of candidates sorted by the employer-side relevance probability $g_j(c)$. The matching $\{(c_1, j_1), (c_2, j_2), (c_3, j_3)\}$ is stable, and it provides a social welfare of $1 + 1 + 0.1 \times 0.1 = 2.01$. Now consider the deterministic ranking policy π so that $\pi(\cdot|c_1)$ recommends j_3 at rank 1, $\pi(\cdot|c_2)$ recommends j_2 at rank 1, and $\pi(\cdot|c_3)$ recommends j_1 at rank 1. This ranking policy provides a social welfare of $1 + 0.9 + 0.9 = 2.8$, but it is not a stable matching (c_1 and j_1 prefer each other over their current match). □

3 OPTIMIZING RANKINGS IN MATCHING MARKETS

We now explore algorithms for computing ranking policies π that maximize social welfare. To motivate the need for these new algorithms, we start by theoretically quantifying the sub-optimality of conventional PRP policies that rank based on one-sided relevances. Denote with π^n the naive relevance-based policy such that for each candidate c, $\pi^n(\cdot|c)$ sorts by $f_c(j)$ and consequently recommends job j at position $\varphi_c(j)$ with probability 1, where $\varphi_c(j)$ denotes the rank of j when ranked by $f_c(j)$. Equivalently, consider its equivalent representation using a doubly-stochastic matrix $\mathbf{M}_C^{\pi^n}$, which we compare against the social welfare optimal ranking $\mathbf{M}_C^{\pi^*}$.

theoremsubopt There exist two-sided matching markets over $|C| = |\mathcal{J}| = n$ with examination models $v(\cdot)$ and relevance probabilities $f_c(j)$ and $g_j(c)$, such that the gap in social welfare between the optimal ranking $\mathbf{M}_C^{\pi^*}$ and the naive one-side relevance-based ranking $\mathbf{M}_C^{\pi^n}$ is larger than $\Theta(n)$. The proof is in Appendix 6.2. The key idea is to construct an instance where there is a popular employer j^* that has high probability of relevance for all candidates. The naive relevance-based ranking will rank j^* at the top position for all candidates and create crowding, while the optimal policy will also consider less crowded employers that may have negligibly smaller relevance probability.

3.1 A Tractable Optimization Objective

Given the sub-optimality of naive rankings, we now develop algorithms that directly optimize social welfare. We first show that the original objective is intractable and hard to optimize directly. To address this, a lower bound is derived and we propose efficient algorithms for solving it.

Before diving into the analysis, we will introduce some notation that will be used in this section. Analogous to $\varphi_c(j)$, we define $\varphi_j(c) \in \{1, 2, 3, \cdots, |C|\}$ as the rank of candidate c when ranking all candidates in C by employer j's relevance probability $g_j(c)$. We break ties arbitrarily. Inversely, we define $\varphi_j^{-1}(s) := \{c' \in C | \varphi_j(c') = s\}$ as the candidate listed in rank s among C when ranked by $g_j(c)$. Correspondingly, we define the priority set for employer j w.r.t. candidate c as $A_j(c) := \{c' \in C | \varphi_j(c') < \varphi_j(c)\}$, which includes the candidates who have higher relevance probability to employer j than candidate c, measured by $g_j(c)$. Based on this, let $F^l_{(j,c)}$ with $l \leq |A_j(c)|$ be the set of all subsets of l items that can be selected from $A_j(c)$: $F^l_{(j,c)} := \{B \subseteq A_j(c) | |B| = l\}$.

Given examination function $v(\cdot)$, and two-sided relevance probabilities $f_c(j)$ and $g_j(c)$, note that our *social welfare* objective can be written as:

$$\mathbf{SW}(\pi) = \sum_{c \in C} \sum_{j \in \mathcal{J}} \mathbb{P}^{\pi}_{c,j} \mathbb{P}^{\pi}_{j,c} = \sum_{c \in C} \sum_{j \in \mathcal{J}} \underbrace{\left(f_c(j) \mathbf{e}_j^T M_c^{\pi} \mathbf{v}\right)}_{\mathbb{P}^{\pi}_{c,j}} \underbrace{g_j(c) \mathbb{E}\left[v\left(\mathbf{rk}_j^{\pi}(c)\right)\right]}_{\mathbb{P}^{\pi}_{j,c}}$$

The probability $\mathbb{P}^{\pi}_{c,j}$ of candidate c applying to employer j is tractably linear in the policy π (or its corresponding doubly stochastic matrices $\mathbf{M}_C^{\pi}$). The difficulty lies in analyzing the term $\mathbf{rk}_j^{\pi}(c)$, which is a random variable that depends on whether other candidates $c \in C$ apply to employer j and the relative relevance among them. The following lemma provides the exact distribution of $\mathbf{rk}_j^{\pi}(c)$, with the parameters depending on the ranking policy π.

Lemma 2. *For any fixed $j \in \mathcal{J}$, $c \in C$, under ranking policy π, the rank $\mathbf{rk}_j^{\pi}(c)$ of candidate c in the set of candidates that applied to employer j is a random variable with $\mathbf{rk}_j^{\pi}(c) \sim 1 + X^{\pi}_{j,c}$, where $X^{\pi}_{j,c}$ is a Poisson Binomial random variable with parameters*

$\big[\mathbb{P}^{\pi}_{\varphi_j^{-1}(1),j}, \mathbb{P}^{\pi}_{\varphi_j^{-1}(2),j}, \cdots, \mathbb{P}^{\pi}_{\varphi_j^{-1}(\varphi_j(c)-1),j}\big]$. *Each element corresponds to the probability of candidate* $c' \in A_j(c)$ *applying to employer* j. *For* $k \in \{1, 2, \cdots \varphi_j(c)\}$, *we have:*

$$\mathbb{P}(\mathbf{rk}_j^{\pi}(c) = k) = \sum_{U \in F^{k-1}_{(j,c)}} \prod_{s \in U} \mathbb{P}^{\pi}_{s,j} \prod_{r \in A_j(c) \setminus U} (1 - \mathbb{P}^{\pi}_{r,j}) \tag{7}$$

Proof. Fix any stochastic ranking policy π, for employer j and candidate c', we use $Y^{\pi}_{c',j}$ to denote the event that candidate c' applies to employer j, and it is easy to see that $Y^{\pi}_{c',j}$ is a Bernoulli RV with parameter $\mathbb{P}(Y^{\pi}_{c',j} = 1) = \mathbb{P}^{\pi}_{c',j}$. Also, the rank of candidate c in employer j's list is 1+ number of candidates who has higher relevance probability to j (i.e., candidates in the set $A_j(c)$) and also applied to employer j, with $\mathbf{rk}_j^{\pi}(c) \overset{\mathcal{D}}{\sim} 1 + \sum_{l=1}^{\varphi_j(c)-1} Y^{\pi}_{\varphi_j^{-1}(l),j}$. Then it is easy to see that the sum of independent Bernoulli trails (but not necessarily identically distributed) follows Poisson Binomial with probabilities $\big[\mathbb{P}^{\pi}_{\varphi_j^{-1}(1),j}, \mathbb{P}^{\pi}_{\varphi_j^{-1}(2),j}, \cdots, \mathbb{P}^{\pi}_{\varphi_j^{-1}(\varphi_j(c)-1),j}\big]$, and the PMF could be easily shown as in Equation 7. □

Given Lemma 2, it is easy to see that the PMF of the distribution of $\mathbf{rk}_j^{\pi}(c)$ involves $n!/((n-l)!l!)$ terms of summation (with $n = |A_j(c)|$ and $l \in \{1, \cdots, \varphi_j(c) - 1\}$), which poses an extensive computational burden. While a recursive formula exists in the literature [32], the complicated form w.r.t. $\mathbb{P}^{\pi}_{c',j}$ poses significant challenges when we treat these probabilities as a function of π (or its $\mathbf{M}^{\pi}_{C}$). To avoid this, we instead optimize the following lower bound on the original objective $\mathbf{SW}(\pi)$, which only requires a mild condition on the examination function and is much easier to compute and optimize.

theoremuti If the examination function $v(\cdot)$ is convex, the following expression lower bounds the social-welfare objective of the stochastic ranking policy π:

$$\underline{\mathbf{SW}}(\pi) := \sum_{c \in C} \sum_{j \in \mathcal{J}} f_c(j) g_j(c) v\big(1 + \sum_{c' \in A_j(c)} f_{c'}(j) \mathbf{e}_j^T M^{\pi}_{c'} \mathbf{v}\big) \mathbf{e}_j^T M^{\pi}_{c} \mathbf{v} \tag{8}$$

Proof. The proof is based on the convexity of v and the simple form of the expectation of the Poisson Binomial random variable.

$$\begin{aligned} \mathbf{SW}(\pi) &= \sum_{c \in C} \sum_{j \in \mathcal{J}} \big(f_c(j) \mathbf{e}_j^T M^{\pi}_c \mathbf{v}\big) g_j(c) \mathbb{E}\big[v\big(\mathbf{rk}_j^{\pi}(c)\big)\big] \\ &\geq \sum_{c \in C} \sum_{j \in \mathcal{J}} \big(f_c(j) \mathbf{e}_j^T M^{\pi}_c \mathbf{v}\big) g_j(c) v\big(\mathbb{E}\big[\mathbf{rk}_j^{\pi}(c)\big]\big) \\ &= \sum_{c \in C} \sum_{j \in \mathcal{J}} f_c(j) g_j(c) v\big(1 + \sum_{c' \in A_j(c)} f_{c'}(j) \mathbf{e}_j^T M^{\pi}_{c'} \mathbf{v}\big) \mathbf{e}_j^T M^{\pi}_c \mathbf{v} \end{aligned} \tag{9}$$

The inequality comes from Jensen's inequality for the convex $v(\cdot)$. The final equality is based on the expectation of the $\mathbf{rk}_j^{\pi}(c)$. □

It is worth noting that convexity of v is not restrictive, since most of the commonly used examination models like $v(x) = 1/x$ and $v(x) = 1/\log_2(1 + x)$ satisfy this condition.

Now we are ready to formulate the main optimization problem, which optimizes the tractable *social-welfare-aware* objective $\underline{\mathbf{SW}}(\pi)$ over the set of doubly stochastic matrices, one for each candidate:

$$\begin{aligned} \underset{\mathbf{M}^{\pi}_{C} := \{M^{\pi}_c\}_{c \in C}}{\text{maximize}} \quad & \sum_{c \in C} \sum_{j \in \mathcal{J}} f_c(j) g_j(c) v\big(1 + \sum_{c' \in A_j(c)} f_{c'}(j) \mathbf{e}_j^T M^{\pi}_{c'} \mathbf{v}\big) \mathbf{e}_j^T M^{\pi}_c \mathbf{v} \\ \text{s. t.} \quad & \mathbf{1}^T M^{\pi}_c = \mathbf{1}^T, M^{\pi}_c \mathbf{1} = \mathbf{1}, \forall c \in C \end{aligned} \tag{10}$$

Multiple approaches can be used to optimize Equation (10). One option is projected gradient descent, where the projection of any positive matrix into the set of doubly stochastic matrices can be computed by the Sinkhorn-Knopp Algorithm [35, 40], which is known to minimize the KL divergence of any nonnegative matrix to the Birkhoff Polytope [40]. An alternative is conditional gradient descent. Since the set of doubly stochastic matrices is convex, we can utilize any convex optimization solver to find the descent direction efficiently. In our experiments, we use the Frank-Wolfe approach (Algorithm 1). For specific examination functions, more specialized algorithms exists. For example, for $v(x) = 1/x$ the optimization problem in Equation (10) becomes a fractional program that can be optimized with an iterative convex-concave procedure.

Algorithm 1: Social-Welfare Optimization via Frank-Wolfe

```
Result: the doubly stochastic matrices: $\mathbf{M}^{\pi}_{C}$
Input: relevance $f_c(\cdot)$ and $g_j(\cdot)$, examination function $v(\cdot)$,
  stopping criterion $\epsilon$, timesteps $T$, learning rate $\eta_t$;
Initialize $M^{\pi}_c = \mathbf{1}\mathbf{1}^T/|\mathcal{J}|, \forall c \in C$;
for $t = 0, 1, \cdots, T$ do
    $S^*_C \in \text{argmin}_{S_C} -\nabla\underline{\mathbf{SW}}(\mathbf{M}^{\pi}_{C})^T S_C$;
            $\mathbf{1}^T S_c = \mathbf{1}^T, S_c\mathbf{1} = \mathbf{1}, \forall c \in C, S_C = \{S_c\}_{c \in C}$;
    $M^{\pi}_{C} \leftarrow (1 - \eta_t)\mathbf{M}^{\pi}_{C} + \eta_t S^*_C$
end
```

4 EXPERIMENTS

In this section, we empirically evaluate several key properties of our approach. We first present experiments on synthetic data which allows us to vary the properties of the two-sided markets to explore the robustness of the method. In addition, we also assess our method on real-world datasets for external validity, including a benchmark dataset from a dating platform, and a new dataset from a virtual-conference networking-recommendation system we built.

4.1 Analysis on Synthetic Data

To examine how our method performs in comparison to baselines over a range of matching markets with different characteristics, we create synthetic datasets as follows. We construct matching markets with n employers and $1.5n$ job candidates to avoid unrealistic symmetry, and other levels of asymmetry would also work here. In the simplest case, we generate relevance probabilities $\tilde{f}_c(j)$ and $\tilde{g}_j(c)$ through independent and uniform draws from $[0, 1]$. We refer to this as the *random* setting, but also consider more structural preferences. One type of structure is *crowding* on some employers and candidates. To create a setting with crowding, we rank employers and candidates in arbitrary order and name them $j_1, j_2, \cdots, j_{|\mathcal{J}|}$ and $c_1, c_2, \cdots, c_{|C|}$. For the candidate-side relevance, we linearly interpolate the relevance probability in $[0, 1]$ and define $\tilde{f}_c(j_i) = 1 - \frac{i-1}{|\mathcal{J}|-1}$ identically for all c so that the relevances of a

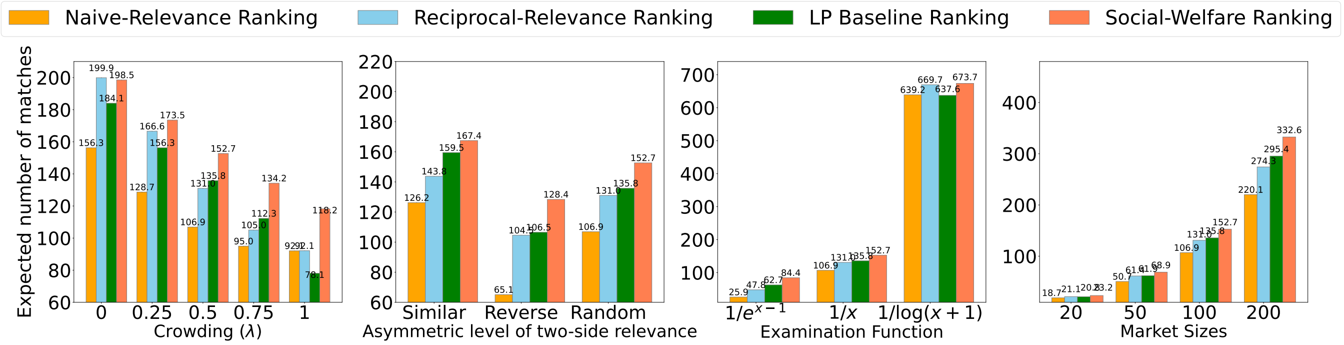


Figure 2: Results on synthetic dataset. The parameters are ($\lambda = 0.5$, random, $v(x) = 1/x$, $n = 100$), unless stated otherwise. The standard errors are on the order of $1e - 2$ and invisible in the graph.

fixed employer are the same to all candidates. For the employer-side relevance, we similarly take $\tilde{g}_j(c_i) = 1 - \frac{i-1}{|C|-1}$ for all j. To adjust the level of crowding, we take the convex combination of the random setting and the fully crowded setting with parameter λ: $f_c(j) := (1-\lambda)\bar{f}_c(j) + \lambda\tilde{f}_c(j)$ and $g_c(j) := (1-\lambda)\bar{g}_c(j) + \lambda\tilde{g}_c(j)$. If not mentioned otherwise, we use $\lambda = 0.5$ in the experiments, as well as a market size of $n = 100$ and the examination function $v(x) = 1/x$ for all candidates and employers. We measure the quality for various ranking policies by the original social welfare objective $\mathbf{SW}(\pi)$, which we estimate using 10000 Monte Carlo simulations of the matching market interaction process described in Section 2. We simulate this process 10 times and report the average result.

We compare our *Social-Welfare Ranking* against the following baselines. *Naive-Relevance Ranking* ranks employers by the one-sided relevance $f_c(j)$ for each candidate c. *Reciprocal-Relevance Ranking* ranks by the reciprocal relevance probability $f_c(j)g_j(c)$ for each candidate c, representing a heuristic objective that accounts for the reciprocal nature of a match while ignoring dependencies between candidates. The closest problem setting from the literature is that of selecting a set of users for recommendation in an online dating platform [38]. While this method only computes sets, we convert the solution into a ranking using the computed probability of a user being included in the recommended set. We call this the *LP Baseline Ranking*. This baseline relies on picking a parameter n_c, which is the maximum number of messages that an applicant could send and receive. To make sure we do not disadvantage this baseline with a poor choice of n_c, we compute the solutions for all $n_c \in \{2, 5, 10, 20, 50, 100\}$ and then pick the solution with the best social welfare in hindsight.

How do the methods perform for different levels of crowding? With crowding we refer to a situation where some employers become overloaded with applications while others get unnoticed. The leftmost graph in Figure 2 shows the expected number of matches for different levels λ of crowding. Especially for high levels of crowding, the *Social-Welfare Ranking* performs substantially better than the baselines. While the *Naive-Relevance Ranking* performs poorly for all levels of crowding, the *Reciprocal-Relevance Ranking* performs equivalent to the *Social-Welfare Ranking* when the preferences are fully random and there is no crowding. However, the *Reciprocal-Relevance Ranking* fails to account for collisions in high-crowding settings, where it does no better than the *Naive-Relevance Ranking*. The *LP Baseline Ranking* consistently underperforms the *Social-Welfare Ranking*, which is not surprising since it optimizes sets and not rankings. It performs particularly poorly under high-crowding levels, where its constraints force lower ranked jobs to have a higher probability of being shown to ensure that candidates/employers do not send/receive too many applications.

How do the methods perform when there is structure in the relevance probabilities? Most real-world problems will contain some structure in the relevance probabilities $f_c(j)$ and $g_j(c)$. We now explore the two complementary cases where the candidate and employer relevance probabilities are either similar to each other or the reverse of each other. To construct *similar* two-sided relevances, we take $\bar{g}_j(c) = \min\{\max\{\bar{f}_c(j) + e, 0\}, 1\}$ with $e \sim \mathcal{N}(0, 0.2)$. For the *reversed* two-sided relevances, we take $\bar{g}_j(c) = \min\{\max\{1 - \bar{f}_c(j) + e, 0\}, 1\}$ with $e \sim \mathcal{N}(0, 0.2)$. For the *random* two-sided relevances, we use the construction already introduced above. The results are shown in the second plot of Figure 2. As the level of asymmetry moves from reverse to similar, the expected number of matches for all methods increases as expected. Moreover, for all relevance structures, *Social-Welfare Ranking* consistently achieves substantially higher social welfare than the baseline methods.

How does the examination function influence the relative performance? The examination function $v(\cdot)$ models how many results people are able or willing to browse. A steep drop-off in examination probability, like $v(x) = 1/e^{x-1}$, means that they are likely to only evaluate the top few results. A flat examination function, such as $v(x) = \frac{1}{\log(1+x)}$, means that they are likely to go further down. The third plot in Figure 2 shows the expected number of matches as we change the examination function. Unsurprisingly, a flatter examination function leads to more matches and little difference between the methods, since results are likely to be discovered no matter where they are placed in the ranking. For the steepest examination function, the advantage of the *Social-Welfare Ranking* over the baselines is largest, and it almost doubles the number of matches compared to the *Reciprocal-Relevance Ranking*.

How does the size of the markets affect the methods? In this experiment, we vary the size of market to understand how this affects the effectiveness of the methods. Results are shown in the rightmost plot of Figure 2. As market size increases, all the ranking methods

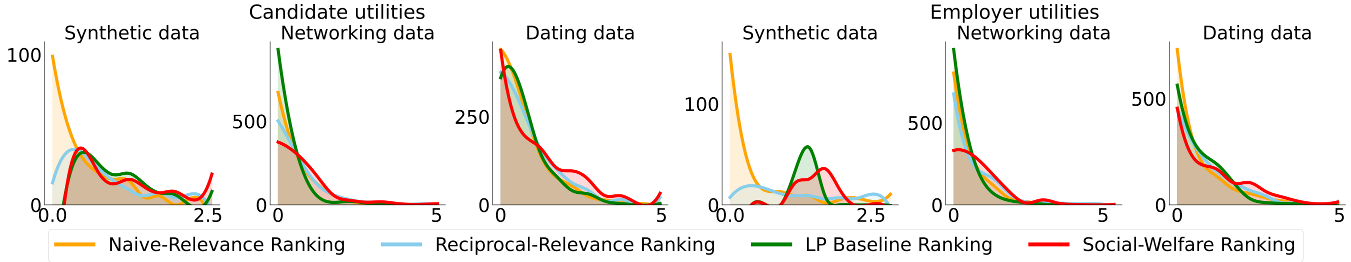


Figure 3: Number of candidates/employers (y-axis) that derive a specific amount of utility (x-axis) from the recommendations.

Table 1: Social Welfare (± two stderr) on real-world datasets.

Dataset	Networking	Online Dating
Naive-Relevance Ranking	604.0 ± 0.11	844.0 ± 0.10
Reciprocal-Relevance Ranking	763.8 ± 0.06	957.2 ± 0.12
LP Baseline Ranking	456.9 ± 0.52	807.9 ± 0.69
Social-Welfare Ranking	824.1 ± 0.18	1199.2 ± 0.14

achieve higher utility, which is expected since there are more opportunities for matches. More interestingly, the relative performance among the methods is largely unaffected by market size.

4.2 Validation on Real-World Data

We also validated our method on two real-world datasets. First, we collected a new dataset[3] by launching a networking recommendation system for a major computer science conference. The goal was to help participants find other participants that they may want to interact with. This recommendation system fits naturally in the two-sided matching markets framework as each user acts proactively by sending messages, scheduling meetings, etc. to other users and the recommendation is successful if the other user replies positively. To account for the fact that each user can serve as both the proactive side (initiate interaction) and the reactive side (reply to the messages), we put each of the 925 users on both sides of the market, with imputed asymmetric two-sided relevances. The imputed relevance is learned by an importance-weighted logistic regression on the observed interaction data. We tested the performance of various ranking algorithms for this data using $v(x) = 1/x$ as the examination function (results for other examination functions follow a similar trend). To reduce computational complexity, we use a two-stage ranking procedure. We first identify the top 100 results based on their reciprocal relevance, and only re-rank those to maximize social welfare. The ranking after the top 100 is by reciprocal relevance. Results over 10 runs are shown in Table 1. The substantial improvement over baseline algorithms verifies that the proposed approach can provide significant benefit in realistic applications. Beyond this overall improvement in social welfare, we find increased individual utility for more than 88% of the participants, which we further discuss in the following section on fairness.

As a second real-world benchmark, we tested our method on data from the online dating service Libimseti [42]. The dataset contains ratings given by a user to other users in the system. We select 500 males and 500 females that have given the most ratings to other users and impute any missing rating using the alternating least squares (ALS) procedure [24], making the simplifying assumption that females will only rate males and vice versa. Again, we evaluate the performance of various algorithms using $v(x) = 1/x$, and use the same two-stage ranking procedure for the *Social-Welfare Ranking*. Results are shown in the third column of Table 1. Again, the *Social-Welfare Ranking* achieves the highest social welfare and exceeds the baseline algorithms by a large margin.

4.3 Impact on Individual Fairness, Adoption and Retention

The issue of fairness is more complex in matching markets than in conventional applications of ranking systems, given the complex dependencies between the individuals. While our objective of maximizing social welfare recognizes the societal importance and impact of many matching markets, we also examined its impact on individual fairness. Figure 3 plots the histograms of individual utilities for different ranking algorithms on both the synthetic and real-world datasets. On all datasets, the *Social-Welfare Ranking* distributes individual utility more equally compared to the baselines. Specifically, the fraction of users in the lowest utility bin is reduced by the *Social-Welfare Ranking*, leading to a more equitable distribution of utility. However, we argue that adding explicit fairness constraints is still useful, and that our optimization-based framework is well-suited for including statistical parity or merit-based exposure constraints [33] in future work. Beyond fairness, we also briefly discuss the need for future work on strategic behavior, such as adoption and retention in Appendix 6.1.

5 CONCLUSIONS

We have formulated the problem of ranking in the two-sided matching markets with the objective of maximizing social welfare in terms of the total number of matches. To make this problem tractable, we identified a lower bound and showed how it can be optimized effectively. This results in the first ranking algorithm that is able to jointly optimize personalized rankings that take into account mutual preferences and limited capacity. In experiments on both synthetic and real-world datasets, we find that the proposed ranking algorithm can consistently achieve the highest social welfare in comparison to existing baselines. Finally, we also outlined directions for future work on strategic behavior and fairness guarantees for ranking systems in matching markets.

Acknowledgements. This research was supported in part by NSF Awards IIS1901168 and IIS-2008139. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

[3] This dataset is available at https://github.com/bayoumi17m/SW-matching-markets

REFERENCES

[1] Aman Agarwal, Ivan Zaitsev, Xuanhui Wang, Cheng Li, Marc Najork, and Thorsten Joachims. Estimating position bias without intrusive interventions. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 474–482, 2019.

[2] Nikolaos D Almalis, George A Tsihrintzis, and Nikolaos Karagiannis. A content based approach for recommending personnel for job positions. In *IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications*, pages 45–49. IEEE, 2014.

[3] Slah Alsaleh, Richi Nayak, Yue Xu, and Lin Chen. Improving matching process in social network using implicit and explicit user information. In *Asia-Pacific Web Conference*, pages 313–320. Springer, 2011.

[4] Itai Ashlagi, Anilesh K Krishnaswamy, Rahul Makhijani, Daniela Saban, and Kirankumar Shiragur. Assortment planning for two-sided sequential matching markets. *arXiv preprint arXiv:1907.04485*, 2019.

[5] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed 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*, pages 2212–2220, 2019.

[6] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*, pages 405–414, 2018.

[7] Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666, 2008.

[8] Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

[9] Ran Duan and Seth Pettie. Linear-time approximation for maximum weight matching. *Journal of the ACM (JACM)*, 61(1):1–23, 2014.

[10] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[11] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[12] Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. Unbiased learning-to-rank with biased feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 781–789, 2017.

[13] Il Yong Kim and OL De Weck. Adaptive weighted sum method for multiobjective optimization: a new method for pareto front generation. *Structural and multidisciplinary optimization*, 31(2):105–116, 2006.

[14] Lei Li and Tao Li. Meet: a generalized framework for reciprocal recommender systems. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 35–44, 2012.

[15] Lydia T Liu, Horia Mania, and Michael Jordan. Competing bandits in matching markets. In *International Conference on Artificial Intelligence and Statistics*, pages 1618–1628. PMLR, 2020.

[16] F Masarani and Sadik S Gokturk. On the existence of fair matching algorithms. *Theory and Decision*, 26(3):305–322, 1989.

[17] Rishabh Mehrotra, Ben Carterette, Yong Li, Quanming Yao, Chen Gao, James Kwok, Qiang Yang, and Isabelle Guyon. Advances in recommender systems: From multi-stakeholder marketplaces to automated recsys. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3533–3534, 2020.

[18] Rishabh Mehrotra, James McInerney, Hugues Bouchard, Mounia Lalmas, and Fernando Diaz. Towards a fair marketplace: Counterfactual evaluation of the trade-off between relevance, fairness & satisfaction in recommendation systems. In *Proceedings of the 27th acm international conference on information and knowledge management*, pages 2243–2251, 2018.

[19] Rishabh Mehrotra, Niannan Xue, and Mounia Lalmas. Bandit based optimization of multiple objectives on a music streaming platform. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3224–3233, 2020.

[20] Tsunenori Mine, Tomoyuki Kakuta, and Akira Ono. Reciprocal recommendation for job matching with bidirectional feedback. In *2013 Second IIAI International Conference on Advanced Applied Informatics*, pages 39–44. IEEE, 2013.

[21] Martin Mladenov, Elliot Creager, Omer Ben-Porat, Kevin Swersky, Richard Zemel, and Craig Boutilier. Optimizing long-term social welfare in recommender systems: A constrained matching approach. *arXiv preprint arXiv:2008.00104*, 2020.

[22] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions. *Mathematical programming*, 14(1):265–294, 1978.

[23] Simone Parisi, Matteo Pirotta, Nicola Smacchia, Luca Bascetta, and Marcello Restelli. Policy gradient approaches for multi-objective sequential decision making. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 2323–2330. IEEE, 2014.

[24] Arkadiusz Paterek. Improving regularized singular value decomposition for collaborative filtering. 2007.

[25] Luiz Pizzato, Thomas Chung, Tomasz Rej, Irena Koprinska, Kalina Yacef, and Judy Kay. Learning user preferences in online dating. In *Proceedings of the Preference Learning (PL-10) Tutorial and Workshop, European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*. Citeseer, 2010.

[26] Luiz Pizzato, Tomasz Rej, Joshua Akehurst, Irena Koprinska, Kalina Yacef, and Judy Kay. Recommending people to people: the nature of reciprocal recommenders with a case study in online dating. *User Modeling and User-Adapted Interaction*, 23(5):447–488, 2013.

[27] Filip Radlinski, Paul N Bennett, Ben Carterette, and Thorsten Joachims. Redundancy, diversity and interdependent document relevance. In *ACM SIGIR Forum*, volume 43, pages 46–52. ACM New York, NY, USA, 2009.

[28] Marco Tulio Ribeiro, Anisio Lacerda, Adriano Veloso, and Nivio Ziviani. Pareto-efficient hybridization for multi-objective recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 19–26, 2012.

[29] Stephen E Robertson. The probability ranking principle in ir. *Journal of documentation*, 1977.

[30] Mario Rodriguez, Christian Posse, and Ethan Zhang. Multiple objective optimization in recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 11–18, 2012.

[31] Alvin E Roth and Elliott Peranson. The redesign of the matching market for american physicians: Some engineering aspects of economic design. *American economic review*, 89(4):748–780, 1999.

[32] BK Shah. Distribution of sum of independent integer valued random-variables. *American Statistician*, 27(3):123–124, 1973.

[33] Ashudeep Singh and Thorsten Joachims. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2219–2228, 2018.

[34] Ashudeep Singh and Thorsten Joachims. Policy learning for fairness in ranking. In *Advances in Neural Information Processing Systems*, pages 5426–5436, 2019.

[35] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

[36] Krysta M Svore, Maksims N Volkovs, and Christopher JC Burges. Learning to rank with multiple objective functions. In *Proceedings of the 20th international conference on World wide web*, pages 367–376, 2011.

[37] Chia-Hsin Ting, Hung-Yi Lo, and Shou-De Lin. Transfer-learning based model for reciprocal recommendation. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 491–502. Springer, 2016.

[38] Kun Tu, Bruno Ribeiro, David Jensen, Don Towsley, Benyuan Liu, Hua Jiang, and Xiaodong Wang. Online dating recommendations: matching markets and learning preferences. In *Proceedings of the 23rd international conference on world wide web*, pages 787–792, 2014.

[39] Saúl Vargas and Pablo Castells. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 109–116, 2011.

[40] Fei Wang, Ping Li, and Arnd Christian Konig. Learning a bi-stochastic data similarity matrix. In *2010 IEEE International Conference on Data Mining*, pages 551–560. IEEE, 2010.

[41] Lequn Wang and Thorsten Joachims. Fairness and diversity for rankings in two-sided markets. *arXiv preprint arXiv:2010.01470*, 2020.

[42] Peng Xia, Benyuan Liu, Yizhou Sun, and Cindy Chen. Reciprocal recommendation system for online dating. In *2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 234–241. IEEE, 2015.

[43] Hongtao Yu, Chaoran Liu, and Fuzhi Zhang. Reciprocal recommendation algorithm for the field of recruitment. *Journal of Information & Computational Science*, 8(16):4061–4068, 2011.

[44] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pages 1171–1180, 2017.

[45] Chongjie Zhang and Julie A Shah. Fairness in multi-agent sequential decision-making. *Advances in Neural Information Processing Systems*, 27:2636–2644, 2014.

[46] Lizi Zhang, Hui Fang, Wee Keong Ng, and Jie Zhang. Intrank: Interaction ranking-based trustworthy friend recommendation. In *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*, pages 266–273. IEEE, 2011.

6 APPENDICES

6.1 Discussion on Adoption and Retention

In this section, we want to at least briefly discuss the need for future work on strategic behavior and fairness guarantees for rankings in matching markets. We posit that these issues are even more important than for ranking systems in conventional markets, since for matching markets the rankings and the actions of all participants shape the utilities of each individual in complex and interdependent ways. This is true for any ranking system applied to matching markets, whether it optimizes social welfare, any other objective, or does not do any explicit optimization at all.

To formalize strategic behavior, we view each candidate $c \in C$ in the market as an independent player that can choose among ranking policies π (e.g., different recommender systems) or equivalently their doubly stochastic matrices M_c^π. For each player c, the payoff $R_c(M_c^\pi, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi'})$ is the tractable lower bound of total expected number of matches that player c could get, which is a function of player c's action M_c^π, and the other players' actions $\bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi'}$.

$$R_c(M_c^\pi, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi'}) = \sum_{j\in\mathcal{J}} f_c(j) g_j(c) v\big(1+\sum_{c'\in A_j(c)} f_{c'}(j)\mathbf{e}_j^T M_{c'}^{\pi'}\mathbf{v}\big)\mathbf{e}_j^T M_c^\pi \mathbf{v}$$

Adoption. The first question we consider is whether candidates will want to participate in the social-welfare optimal system, or whether they will prefer the naive ranking $M_c^{\pi^n}$ which ranks by their own relevance probabilities $f_c(j)$? As strategic agents, candidates will switch to their social-welfare optimal ranking M_c^π, if it increases their utility compared to $M_c^{\pi^n}$. If we assume that during the initial fielding of the system all candidates $c' \in C$ use $M_{c'}^{\pi^n}$, we may want to add the following constraints to our optimization objective, enforcing that all agents have an ϵ-incentive to switch.

$$R_c(M_c^\pi, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi^n}) \geq R_c(M_c^{\pi^n}, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi^n}) + \epsilon \tag{11}$$

It is worth noting that these constraints are linear in M_c^π. Hence, we can simply incorporate them into the Frank-Wolfe Algorithm.

Retention. The second question considers the behavior of the candidates once the system has been widely adopted. In particular, do candidates have an incentive to abandon the system and return to their naive ranking $M_c^{\pi^n}$?

To avoid this, the system ranking M_c^π should provide a larger utility than $M_c^{\pi^n}$ given that all other candidates stay with the system, which could be enforced by adding additional constraints of the form

$$R_c(M_c^\pi, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi}) \geq R_c(M_c^{\pi^n}, \bigcup_{\tilde{c}\in C\setminus c} M_{\tilde{c}}^{\pi}) + \epsilon. \tag{12}$$

Unfortunately, this constraint set is not convex and it is difficult to use Algorithm 1 directly. However gradient descent ascent (GDA) could be used to find the solution of the Lagrangian dual form of the problem.

Experimental Examination. In this experiment, we examine how does the social-welfare optimized ranking affect individual utilities. As discussed in aforementioned paragraphs, the individual utilities of the candidates can affect adoption, retention and fairness of the ranking system. The first row of Figure 4 considers the case where all users switch from using the *Naive-Relevance Ranking* to using the *Social-Welfare Ranking* on the synthetic dataset. It shows a histogram of candidates according to how much they gain from this switch in terms of expected number of matches. Surprisingly, none of the candidates is worse off in this switch for the synthetic dataset. Nevertheless, there are still potential fairness issues and we see on the real-world datasets that this uniformity in gain is not always guaranteed. The second row considers the gain from adoption, which is also non-negative for all candidates on the synthetic dataset and a large fractional of candidates on the networking recommendation dataset. However, the proportion of positive gains on the online dating dataset is pretty low. This calls for interesting future work on incorporating adoption in recommender system design. Finally, the third row shows that all candidates in the synthetic dataset (or most candidates for real-world datasets) are better off staying in the system than switching back to the *Naive-Relevance Ranking*. We also explored other variants of our synthetic data, and generally found that the *Social-Welfare Ranking* is beneficial for most users. This is pretty encouraging for future work, since it suggests that stronger guarantees on individual utilities may be achievable with little drop in social welfare.

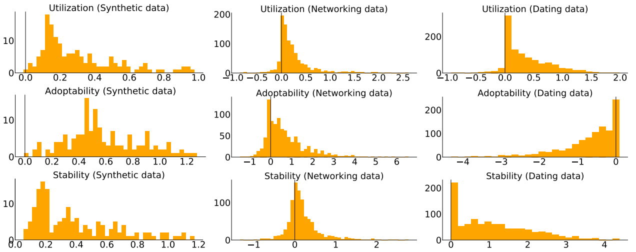


Figure 4: Histogram of individual utility gains on synthetic data with $\lambda = 0.5$, random, $v(x) = 1/x$ and $n = 100$ (left column) and two real-world datasets: Networking Recommendation (middle column) and Online Dating (right column). (Top) Gain when all users switch from *Naive-Relevance Ranking* to *Social-Welfare Ranking*. (Middle) Gain from adopting *Social-Welfare Ranking* even though others are not. (Bottom) Gain from not abandoning *Social-Welfare Ranking* if all others keep using it.

6.2 Proof of Theorem 3

The $\Theta(n)$ gap in Theorem 3 is achieved by identifying the specific instance of the two-sided matching market (particular choice of relevance probability and examination model). Consider the following two-sided market: there are n employers (denoted as $\{j_1, j_2, \cdots, j_n\}$) and n candidates (denoted as $\{c_1, c_2, \cdots, c_n\}$), with $n \geq 2$. On the candidate side, they have highly correlated relevance probability over jobs, while each job has different relevance probability over candidates. The employers' relevance probability ordering (ordered by descending order of $g_j(\cdot)$) is given by the left circulant matrix and the top 3 employers of the candidates' relevance probability list (ordered by descending order of $f_c(\cdot)$) is given in the right table:

$$\begin{array}{c|ccccc} j_1 & c_1 & c_2 & c_3 & \cdots & c_n \\ \hline j_2 & c_2 & c_3 & \cdots & c_n & c_1 \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ j_k & c_k & \cdots & \cdots & c_{k-2} & c_{k-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline j_n & c_n & c_1 & \cdots & c_{n-2} & c_{n-1} \end{array} \qquad \begin{array}{c|ccc} c_1 & j_1 & j_2 & j_3 \\ \hline c_2 & j_1 & j_2 & j_3 \\ \hline \vdots & \vdots & \vdots & \vdots \\ c_k & j_1 & j_k & j_{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ \hline c_n & j_1 & j_n & j_{n-1} \end{array} \quad \text{and}$$

for each candidate c_i, for the employers j that not in the top 3 position of c_i's relevance list, we set arbitrary $f_{c_i}(j) \in [0, f_{c_i}(\varphi_{c_i}^{-1}(3))]$. Given this, we define the cardinal relevance probability $f_c(\cdot)$ and $g_j(\cdot)$ from the above ordinal ordering as:

$$f_{c_i}(j_k) = \frac{n - (\varphi_{c_i}(j_k) - 1)}{n} \qquad g_{j_k}(c_i) = \frac{n - (\varphi_{j_k}(c_i) - 1)}{n}$$

The examination model E_m in this case is given by $v(x) = 0.1^{x-1}$, if $x \leq m$ and 0 otherwise. This mimics the scenario that even as the market size n grows, people tends to only examine the top m recommendations due to time and resource constraints. Here we choose $m = 2$ for simplicity in the proof, but it could be generalized to any fixed m. As candidates only examine the top 2 positions, only the top 2 positions in our rankings do matter. Therefore for all rankings we compare in the proof, we only list the top 2 positions in the ranking. The naive-relevance based ranking π^n only recommends the employer to candidates by their relevance to the candidate. Therefore π^n is a deterministic policy based on the following permutation σ^n:

$$\begin{aligned} &rank(j_1|\sigma^n(c_1)) = 1 \qquad && rank(j_2|\sigma^n(c_1)) = 2 \\ &rank(j_1|\sigma^n(c_i)) = 1 \qquad && rank(j_i|\sigma^n(c_i)) = 2 \ \ \forall i \geq 2 \end{aligned} \tag{13}$$

Now we analyze the utility $\mathbf{SW}(\pi^n)$, and it is worth noting that the employers will only examine the top 2 positions. Then we have:

$$\begin{aligned} \mathbf{U}_{c_1}(\pi^n) &= \mathbb{P}(Y_{c_1,j_1}^{\pi^n}) + \mathbb{P}(Y_{c_1,j_2}^{\pi^n}) \\ &= 1 + \frac{0.1(n-1)}{n} \times \frac{1}{n}\left(1 - \frac{0.1(n-1)}{n} + 0.1\frac{0.1(n-1)}{n}\right) \\ &\leq 1.1 + \frac{0.9}{n} \\ \mathbf{U}_{c_2}(\pi^n) &= \frac{0.1(n-1)}{n} + \frac{0.1(n-1)}{n} = \frac{0.2(n-1)}{n} \end{aligned} \tag{14}$$

For candidate c_i with $i \geq 3$, we have

$$\mathbf{U}_{c_i}(\pi^n) = \mathbb{P}(Y_{c_i,j_1}^{\pi^n}) + \mathbb{P}(Y_{c_i,j_i}^{\pi^n}) = \frac{0.1(n-1)}{n} \tag{15}$$

Then the social welfare generated by policy π^n is:

$$\mathbf{SW}(\pi^n) \leq 1.1 + \frac{0.9}{n} + \frac{0.2(n-1)}{n} + \frac{0.1(n-1)(n-2)}{n} \leq 1 + 0.1n + \frac{1}{n} \tag{16}$$

Now we consider a different deterministic policy π^s based on the following permutation σ^s:

$$\begin{aligned} &rank(j_1|\sigma^s(c_1)) = 1 \qquad && rank(j_2|\sigma^s(c_1)) = 2 \\ &rank(j_i|\sigma^s(c_i)) = 1 \qquad && rank(j_{i-1}|\sigma^s(c_i)) = 2 \ \ \forall i \geq 2 \end{aligned} \tag{17}$$

Similarly we analyze the utility $\mathbf{SW}(\pi^s)$ and we have:

$$\begin{aligned} &\mathbf{U}_{c_1}(\pi^s) = \mathbb{P}(Y_{c_1,j_1}^{\pi^s}) + \mathbb{P}(Y_{c_1,j_2}^{\pi^s}) \\ &= 1 + \frac{0.1(n-1)}{n} \times \frac{1}{n}\left(1 - \frac{(n-1)}{n} + 0.1\frac{(n-1)}{n}\right) \geq 1 + \frac{0.01(n-1)}{n^2} \end{aligned} \tag{18}$$

For candidate c_2, we have

$$\mathbf{U}_{c_2}(\pi^s) = \mathbb{P}(Y_{c_2,j_2}^{\pi^s}) + \mathbb{P}(Y_{c_2,j_1}^{\pi^s}) = \frac{(n-1)}{n} + \frac{0.1^2(n-1)}{n} = \frac{1.01(n-1)}{n} \tag{19}$$

For candidate c_i with $i \geq 3$, we have

$$\begin{aligned} &\mathbf{U}_{c_i}(\pi^s) = \mathbb{P}(Y_{c_i,j_i}^{\pi^s}) + \mathbb{P}(Y_{c_i,j_{i-1}}^{\pi^s}) \\ &= \frac{(n-1)}{n} + \frac{0.1(n-2)}{n}\left(1 - \frac{n-1}{n} + \frac{0.1(n-1)}{n}\right) \geq \frac{n-1}{n} \end{aligned} \tag{20}$$

Then the social welfare generated by policy π^s is:

$$\begin{aligned} \mathbf{SW}(\pi^s) &\geq 1 + \frac{0.01(n-1)}{n^2} + \frac{1.01(n-1)}{n} + \frac{(n-1)(n-2)}{n} \\ &\geq n - 0.99 + \frac{0.99}{n} \end{aligned} \tag{21}$$

Now we denote the optimal solution for social welfare objective as π^*. Therefore, we have $\mathbf{SW}(\pi^*) - \mathbf{SW}(\pi^n) \geq \mathbf{SW}(\pi^s) - \mathbf{SW}(\pi^n) = \Theta(n)$.

6.3 Additional Details on Dataset and Implementation

Datasets. We calculate the two-sided relevances in the networking recommendation dataset as follows. In the recommendation phrase, the recommendation system estimated relevance scores between all registered users based on features likes similarity of published articles, past co-authorship, past citations, etc. For each user, the system recommended a ranked list of 150 participants, and we collected different forms of positive interactions between them (such as thumb up, send a message, schedule a meeting). In the relevance imputation phrase, we use an importance-weighted (to de-bias position bias) logistic regression with L_2 regularization to learn the relevance function, which allows us to impute directional relevance probabilities between all users. Curiously, the relevance probabilities have substantial directionality, with faculty and postdocs being "crowded". However, the expected number of individuals that each user reaches out is rather uniform between groups. For the online dating dataset , the license could be accessible at http://web.archive.org/web/20180402034337/http://www.occamslab.com/petricek/data/.

Implementation. For all experiments, the relevance-based ranking is a deterministic ranking based on descending order of $f_c(j)$, and the reciprocal-based ranking is a deterministic ranking based on descending order of $f_c(j)g_j(c)$. For our algorithm, *Social-Welfare Ranking*, we use Algorithm 1 with 50 timesteps, stopping criterion 10^{-3}, and constant learning rate $\eta_t = 0.2$. We also tested a decaying learning rate $\eta_t = \frac{1}{t+2}$ and the performance was similar to the constant learning rate. The convex solver we use for finding the gradient direction within Algorithm 1 is CVXPY[8] with SCS. All experiments are performed on a Linux computer cluster.