# Optimizing Long-term Social Welfare in Recommender Systems: A Constrained Matching Approach

**Martin Mladenov** [1] **Elliot Creager** [2 3 *] **Omer Ben-Porat** [4 †] **Kevin Swersky** [1] **Richard Zemel** [2 3] **Craig Boutilier** [1]

## Abstract

Most recommender systems (RS) research assumes that a user's utility can be maximized independently of the utility of the other agents (e.g., other users, content providers). In realistic settings, this is often not true—the dynamics of an RS ecosystem couple the long-term utility of all agents. In this work, we explore settings in which content providers cannot remain viable unless they receive a certain level of user engagement. We formulate the recommendation problem in this setting as one of *equilibrium selection* in the induced dynamical system, and show that it can be solved as an *optimal constrained matching* problem. Our model ensures the system reaches an equilibrium with maximal social welfare supported by a sufficiently diverse set of viable providers. We demonstrate that even in a simple, stylized dynamical RS model, the standard *myopic* approach to recommendation—always matching a user to the best provider—performs poorly. We develop several scalable techniques to solve the matching problem, and also draw connections to various notions of user regret and fairness, arguing that these outcomes are fairer in a utilitarian sense.

## 1. Introduction

Investigations of various notions of *fairness* in machine learning (ML) have shown that, without due care, applying ML in many domains can result in biased outcomes that disadvantage specific individuals or groups (Dwork et al., 2012; Barocas & Selbst, 2016). Content *recommender systems (RSs)*, which match users to content (e.g., news, music, video), typically rely on ML to predict a user's interests to recommend "good" content (Konstan et al., 1997; Jacobson et al., 2016; Covington et al., 2016). Since these predictions are learned from past behavior, many issues of ML fairness arise in RS settings (Beutel et al., 2019).

One aspect of "fairness" that has received little attention emerges when one considers the *dynamics of the RS ecosystem*. Both users and content providers have particular incentives for engaging with an RS platform—incentives which interact, via the RS *matching policy*, to couple the long-term utility of agents on both sides of this content "marketplace." Some work has looked at the impact of RS policies on provider welfare (Singh & Joachims, 2018), on long-term user and provider metrics (e.g., using RL (Chen et al., 2018; Ie et al., 2019b) or assessing various other phenomena (Ribeiro et al., 2020; Celma, 2010)). However, little work has looked at the interaction of the two on the ecosystem dynamics induced by the RS policy (though there are some exceptions, e.g., (Ben-Porat & Tennenholtz, 2018), which are discussed below).

In this work, we focus on *provider* behavior using a stylized model of a content RS ecosystem in which providers require a certain degree of user engagement (e.g., views, time spent, satisfaction) to remain *viable*. This required degree (or "threshold") of engagement reflects their incentives (social, economic, or otherwise) to participate in the RS; and if this threshold is not met, a provider will withdraw from the platform (i.e., their content is no longer accessible). If this occurs, user segments for whom that provider's content is ideal may be disadvantaged, leading such users to derive less utility from the RS. This may often arise, say, for users and providers of *niche* content.

Typical RS policies are *myopic*: given a user request or query, it returns the provider that is (predicted to be) best-aligned with that query. In our model, myopic policies often drive the dynamical system to a poor equilibrium, with low *user social welfare* and poor provider diversity. By contrast, a more *holistic* approach to matching requests to providers can derive much greater user welfare. We formulate policy optimization in our model as a constrained matching problem, which *optimizes for the socially optimal equilibrium of the ecosystem*. We develop scalable techniques for

[*]Work done as a student researcher at Google Research [†]Work partially done as an intern at Google Research [1]Google Research [2]University of Toronto [3]Vector Institute [4]Technion. Correspondence to: Martin Mladenov <mmladenov@google.com>.

*Proceedings of the $37^{th}$ International Conference on Machine Learning*, Online, PMLR 119, 2020. 

computing these policies, and show empirically that they produce much higher social welfare than myopic policies, *even in this relatively stylized model*. Such policies also lead to greater provider diversity—even though our objective only involves *user* utility—by (implicitly) "subsidizing" some providers that would not remain viable under the myopic policy. We examine tradeoffs between user regret and social welfare in this model, showing that user maximum regret tends to be quite low, especially with utility functions exhibiting diminishing returns. Finally, we draw connections to notions of ML fairness and argue that the outcomes induced by our matching-based policies are fairer in a utilitarian sense.

## 2. Challenges for Myopic Content Matching

We first introduce an abstract, but general, formalization of dynamic content RSs. Two key elements drive ecosystem utility and dynamics. First, users derive (possibly nonlinear) utility from the *collection* or *sequence* of content recommended to them, not just from the individual items. Second, providers require minimum levels of user engagement to remain *viable*, i.e., incentivized to engage with the RS. Within this model, we show how *myopic* RS policies often poorly serve both users and providers, and discuss policy types that overcome this.

### 2.1. A Formalization of Dynamic Recommendations

We assume an RS (or platform) that matches *users* $\mathcal{U}$ to *content providers* $\mathcal{C}$. Users issue *queries* for desired content, drawn from some space $\mathcal{Q}$. We assume $\mathcal{U}, \mathcal{C}, \mathcal{Q}$ are finite. Given user $u$'s query $q_u$, the RS returns a provider $c$ from which $u$ derives *immediate reward* $r(q_u, c)$ reflecting match quality. We often assume the existence of some *latent space* $X \subseteq \mathbb{R}^d$ that is used to represent both content and queries. This is common, say, in collaborative filtering (CF), where $X$ is an embedding space constructed by matrix factorization (Salakhutdinov & Mnih, 2007) or neural CF (He et al., 2017; Beutel et al., 2018)). For $c, q_u \in X$ we let $r(q_u, c) = q_u^T c$ be immediate reward.

User queries are received asynchronously and immediately matched to a provider, giving a discrete-event dynamical system over time periods $1, 2, \ldots, t, \ldots$ At each time $t$, a user $u[t]$, drawn from some distribution $\rho(\mathcal{U})$, issues a query $q_u[t] \in X$, itself drawn from distribution $P_{u[t]}(X)$ reflecting that user's interests. The RS *matches* $q_u[t]$ to some provider $c[t] \in \mathcal{C}$, and $u[t]$ derives reward $r(q_u[t], c[t])$.[1] We assume the RS has *complete knowledge* of the location in $X$ of $c$ and $q_u$, and that these embeddings do not change over time. While unrealistic in practice—most RSs continually update user and content representations, and may engage in active exploration to help assess user interests—the problems we address under this assumption are further exacerbated by incomplete information. We discuss this further below.

Let $h_t = ((u[i], q_u[i], c[i]))_{i \leq t}$ be a length $t$ history of past queries and recommendations, $H[t]$ be the set of all such histories, and $H[*] = \cup_{t<\infty} H[t]$. An RS *policy* $\pi : H[*] \times (\mathcal{Q}, \mathcal{U}) \to \Delta(\mathcal{C})$ maps a history and a query to a distribution $\pi(h_t, q_u[t])$ over providers.

Generally, an RS aims to maximize user engagement. We assume that over long horizons, user engagement is optimized by maximizing *user utility*, which in turn drives sustained user satisfaction with the platform. We assume that $u$'s utility is a (possibly non-linear) function $f$ of the reward sequence $\mathbf{r}_u$ obtained over some horizon. For instance, this might be the cumulative sum of rewards; recency-weighted cumulative reward; or a function (e.g., sigmoid) that captures various behavioral phenomena, such as decreasing marginal returns. We discuss several such functions in Sec. 3. However, even with this user focus, the RS must also address the provider incentives—in our case, viability—since providers offer the quality content needed for a thriving ecosystem. Each provider $c$ has a *viability threshold* $\nu_c$ over the amount of (possibly recency-weighted) user engagement (e.g., visits, time spent) generated for it by the RS. Periodically, $c$ compares its overall engagement against $\nu_c$ and abandons the platform—perhaps stochastically–if this threshold is not reached (we detail specific forms of this function below). If a provider becomes unviable, it can no longer be matched by the RS to any user query.

### 2.2. Suboptimality of Myopic Policies

Before developing our methods, we use two simple examples to illustrate why typical *myopic* recommendation policies may be suboptimal.

Typical RSs behave *myopically*: when query $q_u$ is received, it is matched to the provider giving maximum user reward $c^*_{q_u} = \arg\max_{c \in \mathcal{C}} r(q_u, c)$. In the example in Fig. 1(a), each provider has a viability threshold of 2 and is viable in the initial period. The myopic policy first matches each of $u_1, u_2, u_3$ to $c_1$; $u_4, u_5$ to $c_2$; and $u_6$ to $c_3$. This gives total immediate reward $10 + 2\varepsilon$ according to the reward function $r(u, c) = 2 - |u - c|$ for each recommendation. However, since $c_3$ receives only one user, it is no longer viable at the next period. Hence, at all subsequent periods, the RS must match $u_6$ to $c_2$ (with reward 0), attaining a long-run per-period average reward of $8 + 2\varepsilon$.

A *non-myopic policy* can obtain a long-run average reward of $10 - 2\varepsilon$ by matching $u_3$ to $c_2$ and $u_5$ to $c_3$ at each period.

[1] For ease of exposition, we do not distinguish the different content *items* offered by provider $c$. Nothing fundamental changes in our approach if we match queries to specific items if each item is associated with a provider.

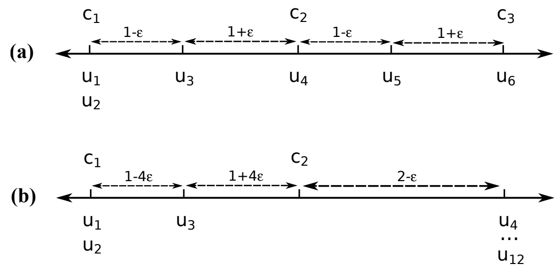



*Figure 1.* Two 1D examples: (a) User $u_i$'s reward for being matched to provider $c_j$ is 2 less the distance between them (e.g., $u_1$ has reward 2 for $c_1$; $u_3$ has reward $1+\varepsilon$ (resp., $1-\varepsilon$) for $c_1$ (resp., $c_2$). We equate reward and utility, and a user with her query. assume that each user issues a single query per period, and that each provider requires 2 user impressions in each period to remain viable at the next period. (b) Similar to (a) except that $c_1$ requires 2 impressions and $c_2$ requires 10.

Under this policy, $c_3$ remains viable, allowing $u_6$ to receive reward 2 (rather than 0) in perpetuity. This comes at a small price to $u_3$ and $u_5$, each of whom receive $2\varepsilon$ less per period. This matching *subsidizes* $c_3$ by matching its content to $u_5$ (who would slightly prefer provider $c_2$). This subsidy leaves $c_2$ vulnerable, so it too is subsidized by the match with $u_3$. Indeed, this matching is *optimal* for any horizon of at least two periods—its average-per-period *user social welfare* (or total reward) is maximized. The maximum loss of utility experienced by any user at any period w.r.t. the myopic policy is quite small, only $2\varepsilon$ (by both $u_3, u_5$)—this is the *maximum (user) regret* of the policy. Finally, this policy keeps all providers viable in perpetuity; the set of viable providers $V = \mathcal{C}$ is an *equilibrium* of the dynamical system induced by the policy. By contrast, the myopic policy reaches an equilibrium $V' = \{c_1, c_2\}$ that has fewer viable providers.

Consider now a policy that matches $u_3$ to $c_3$ at each period, but otherwise behaves myopically. This induces the same equilibrium $V = \mathcal{C}$ as the optimal policy by subsidizing $c_3$ with $u_3$. However, this policy—though improving $u_5$'s utility by $2\varepsilon$ (and her regret to 0)—gives a reward of 0 to $u_3$ (whose regret is $1+\varepsilon$). This policy not only has higher max regret, it also has significantly lower welfare of $9+\varepsilon$.

While not the case in this example, the policy that optimizes social welfare need not minimize max regret. Fig. 1(b) considers a case where the viability threshold differs from each of the two providers. The myopic policy initially matches $\{u_1, \ldots, u_3\}$ to $c_1$ and $\{u_4, \ldots, u_{12}\}$ to $c_2$, after which $c_2$ is no longer viable (and $\{u_4, \ldots, u_{12}\}$ receive no further reward). Thus per-period reward is $5+4\varepsilon$ (and max regret is $\varepsilon$.) The welfare-optimal policy subsidizes $c_2$ by matching $u_3$, increasing welfare marginally by $\varepsilon$ to $5+5\varepsilon$, but also increasing max regret (see $u_3$) to $8\varepsilon$. This illustrates the trade-off between social welfare maximization and max-regret minimization.

These examples show that maximizing user social welfare often requires that the RS take action to *ensure the long-run viability of providers*. The example from Fig 1(a) shows that such considerations need not be explicit, but simply emerge as a *by-product of maximizing user welfare alone*. This also promotes diversity among viable providers that can in some sense be interpreted as being "more fair" to the user population. In particular, it creates a smaller gap between the (long-term) utility values attained by different users across the spectrum of possible topic interests. However, as with provider diversity, this type of fairness is not part of the *explicit* objective that drives the RS policy—rather it is implicit, with fairness emerging as a consequence of trying to maximize *overall* user welfare. We discuss connections to work on ML fairness further below. For a richer illustration of this, see Fig. 2.

### 2.3. Matching Optimization for Recommendation

We now formalize our objectives and optimization approach. For ease of exposition, we assume an *epoch-based* decision problem: time steps are grouped into epochs of fixed length $T$, with user utility and provider viability both determined at the end of each epoch. Let $Q$ denote the induced distribution over queries during any epoch (since user behavior is stationary, so is $Q$). Other forms of user/provider evaluation do not impact the qualitative nature of our results—some require different forms of analysis and optimization, while others carry through easily. For example, if providers use recency-weighted engagement, no substantial changes are needed; but if their evaluation occurs on a continual (not epoch-based) basis, more intricate equilibrium analysis is required and optimization becomes more online in nature.

A policy $\pi$ induces a stochastic dynamical system over a state space, where the state encodes user utility and provider viability at the end of each epoch. Let random variable (RV) $e_t^\pi(c)$ be provider $c$'s engagement at time $t$ under $\pi$, and $E_k^\pi(c)$ its cumulative engagement during epoch $k \geq 1$. If $E_k^\pi(c) \geq \nu_c$, $c$ remains viable at epoch $k+1$, otherwise it abandons the platform. Let $V_k^\pi$ be the set of providers that are viable at the end of epoch $k$, We assume $V_0^\pi = \mathcal{C}$.

Let (RV) $\mathbf{r}_k^\pi(u)$ be user $u$'s reward sequence in epoch $k$ under $\pi$, and $U_k^\pi(u) = f(\mathbf{r}_k^\pi(u))$ be $u$'s utility. *Social welfare* generated by $\pi$ at epoch $k$ is $SW_k^\pi = \sum_{u \in \mathcal{U}} U_k^\pi(u)$. (Long-run) *average social welfare* is $SW_\infty^\pi = \lim_{k\to\infty} [\sum_k SW_k^\pi]/k$. The average utility $U_\infty^\pi(u)$ of $u$ under $\pi$ is defined analogously. If $\pi_u^*$ is the policy that maximizes $u$'s average utility, then $u$'s *regret* under $\pi$ is $Rgrt^\pi(u) = U_\infty^{\pi_u^*}(u) - U_\infty^\pi(u)$. The *maximum regret* of $\pi$ is $MR(\pi) = \max_{u \in \mathcal{U}} Rgrt^\pi(u)$. Let $MC(u) \subseteq \mathcal{C}$ be those providers matched by the myopic policy to queries $\mathcal{Q}_u$ with positive support in $P_u$ when all providers are vi-

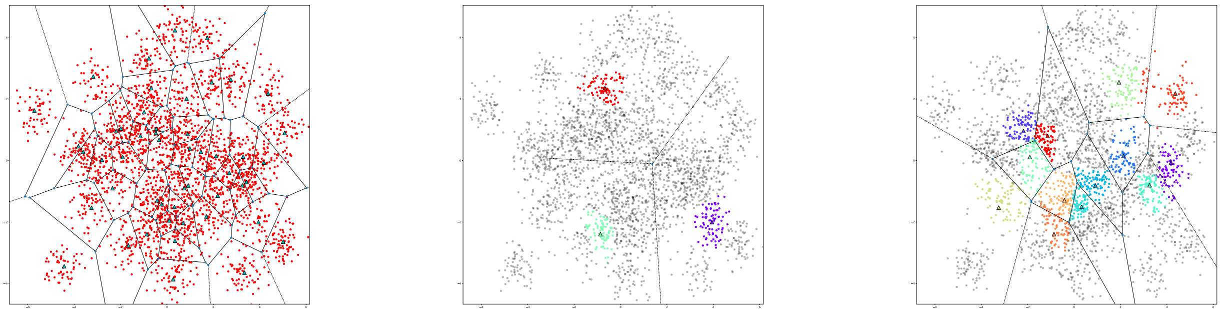


*Figure 2.* Recommendation using dynamics-informed constraints keeps more content providers viable in the long run, improving social welfare amongst users. The first panel shows providers (blue triangles) and users (red dots) embedded in a 2D topic space. The second and third panel respectively show equilibrium under Myopic and proposed linear program (LP) recommendation; here users whose most-preferred provider has dropped out of the ecosystem are shaded in grey, while the remaining users are colored according to their most-preferred provider.

able. Under the mild assumption that, for any $u$, there is a policy that keeps $MC(u)$ viable, $U_\infty^{\pi_u^*}(u) = \mathbb{E}[f(\mathbf{r}^*(u))]$ is a *constant* where $\mathbf{r}^*(u)$ is the realized reward sequence when any $q_u$ generates reward $\arg\max_c r(q_u, c)$.

Our interest in long-run performance leads to a focus on policy behavior in equilibrium w.r.t. provider viability. We say $V \subseteq \mathcal{C}$ is an *equilibrium* of $\pi$ if, for some $k \geq 0$, $V_{k+j}^\pi = V$ for all $j \geq 0$ (i.e., the providers $V_k^\pi$ that are viable after epoch $k$ remain viable in perpetuity). Since most large-scale RSs have massive numbers of users, we assume that the number of queries received at each point in $X$ (or within a suitable set of small subregions of $X$) during each epoch is exactly its expectation. This can be justified by appeal to the law of large numbers for epochs of sufficient duration.[2]

Under these assumptions, the recommendation policy that maximizes average social welfare has an especially simple form. Specifically, there is an optimal *stationary policy* (w.r.t. epochs) which can be formulated as an optimal matching problem under viability constraints. Consider the following *single-epoch decision problem*:

$$\max_\pi \sum_{u\in\mathcal{U}} \mathbb{E}[f(\mathbf{r}^\pi(u))|\pi] \tag{1}$$

$$\text{s.t. } \pi_{q,c} > 0 \text{ only if } \sum_{u,q_u} \overline{Q}(q_u)\pi_{q_u,c} \geq \nu_c, \;\; \forall\, q, c \tag{2}$$

Here $\pi$ is a vector of matching variables $\pi_{q_u,c}$, denoting the proportion of queries $q_u$ to match to $c$.[3] $\overline{Q}(q_u)$ is the expected number of queries of the type $q_u$, but note that the method holds for any non-negative function of $q_u$. The expectation of $u$'s utility $f$ is taken w.r.t. user activation at each $t$ in the epoch, the user query distribution and $\pi$; i.e., the $t$-th component of $\mathbf{r}^\pi(u)$ is distributed as

$$r_t^\pi \sim \rho(u) \sum_{q_u\in\mathcal{Q}} P_u(q_u) \sum_c \pi_{q_u,c} r(q_u, c). \tag{3}$$

Objective (1) optimizes social welfare over the epoch, while constraint (2) ensures that any *matched* provider remains viable at the end of the epoch.

We show that applying this single-epoch policy $\pi$ across all epochs is guaranteed to optimize average user social welfare. Since the expectation $\overline{Q}(q_u)$ is realized exactly, $\pi$ induces an equilibrium during the first epoch. Moreover, the optimization ensures it has maximum welfare, i.e., is the optimal stationary policy. Finally, while a non-stationary $\pi'$ may initially improve welfare relative to $\pi$, its equilibrium welfare cannot exceed that of $\pi$; so any welfare improvement is transient and cannot increase (long-run) average welfare. The stationary $\pi$ given by (1) *anticipates the equilibrium it induces* and only matches to providers that are viable in that equilibrium. This obviates the need to consider more complex policies based on the underlying Markov decision process (we discuss richer policy classes below).

## 3. Solving the Matching Optimization

We now develop several practical methods for solving the optimization (1) to generate optimal policies. We consider formulations that accommodate various forms of user utility: simple linear, cumulative reward; a discounted model that reflects decreasing marginal returns; and non-linear models (of which sigmoidal utility is a motivating example). We

[2]We discuss relaxations of this assumption below.

[3]This can also be interpreted as a stochastic policy, which is the natural practical implementation. When user utility is time-dependent, we sometimes allow the policy to be non-stationary *within* the epoch, writing $\pi_{q_u,c,t}$ for $t \leq T$.

then show how social welfare and regret can be traded off, and briefly describe how the models can be made more robust to uncertainty in the user query stream.

### 3.1. Additive Utility: A Linear Programming Model

We first develop a linear programming (LP) model that applies when user utility is suitably additive, capturing both cumulative reward and a *discounted engagement* model that reflects a natural form of decreasing marginal return.

Let $\alpha \in \mathbb{R}_+^T$ be a vector of non-negative weights. We assume linear user utility $f_\alpha : \mathbf{r} \mapsto \sum_{t=1}^T \alpha_t \mathbf{r}_t$, where user utility in an epoch is the $\alpha$-weighted sum of immediate rewards (cumulative reward is a special case with all $\alpha_t = 1$). Moreover, we consider a class of non-stationary policies that take time itself as their only history feature, $\pi_{h_t,q_u,c} = \pi_{t,q_u,c}$ (if we allow $\pi$ to depend on arbitrary statistics, the problem is a full POMDP).

The expected welfare of $\pi$ over the epoch is

$$\sum_u \mathbb{E}[f_\alpha(\mathbf{r}^\pi(u))|\pi] = \sum_{t=1}^T \sum_{u\in\mathcal{U}} \sum_{q_u\in\mathcal{Q}} \sum_{c\in\mathcal{C}} \alpha_t \pi_{t,q_u,c} \bar{r}(q_u,c),$$

where $\bar{r}(q_u, c) = \rho(u)P_u(q_u)r(q_u, c)$. Since (per-epoch) social welfare $SW^\pi$ is a linear function of $\pi$, Eq. (1) can be reformulated as a mixed-integer linear program (MILP):

$$\begin{aligned}
\max_{\pi,\, y} \quad & \sum_{t=1}^T \sum_{u\in\mathcal{U}} \sum_{q_u\in\mathcal{Q}} \sum_{c\in\mathcal{C}} \alpha_t \pi_{t,q_u,c} \bar{r}(q_u,c) \\
\text{s.t.} \quad & \\
& \sum_c \pi_{q_u,c,t} = 1 && \forall t \in [1:T], q_u \in \mathcal{Q}, \\
& \pi_{q_u,c,t} \le y_c && \forall t \in [1:T], q_u \in Q, c \in \mathcal{C}, \\
& \sum_{u,q_u,t} \overline{Q}(q_u)\pi_{q_u,c,t} \ge \nu_c y_c && \forall c \in \mathcal{C}
\end{aligned} \tag{4}$$

where matching variables $\pi_{t,q_u,c}$ represent the stochastic policy, and provider-viability variables $y_c$ are in $\{0,1\}$ (for cumulative reward, dependence of $\pi$ on $t$ can be removed). Problem (4) is akin to *max-utility constrained facility location*, where user-query-time tuples act as customers and providers act as facilities. Related problems have been investigated in various forms (An et al., 2017; Li, 2019; Cornuejols et al., 1977). All variants (including ours) have basic facility location as a special case (where the constraints are trivial (i.e., $\nu_c = 0$) and are thus NP-hard. Even though their formulations are similar, each has different approximation properties. The combination of objective and constraints we consider has not, we believe, been studied in the literature.

Our problem can be approximated in polynomial time up to a constant factor:

**Theorem 1.** *Problem (4) can be approximated up to factor $\frac{1}{e}$ in polynomial time.*

The core of the proof (see Appendix A.1) is to consider the problem of computing *maximum constrained welfare*, $cSW(C)$, given a *fixed* set $C \subseteq \mathcal{C}$ of viable providers. $cSW(C)$ is the maximum of Eq. (4) when the provider variables are "set" as $y_c = 1$ iff $c \in C$; i.e., we find the best stochastic matching given that all and only providers in $C$ remain viable. With the integer variables removed, the problem becomes a polynomially sized LP. We show that $cSW$ is submodular in the provider set $C$, i.e., $cSW(C \cup \{c', c\}) - cSW(C \cup \{c'\}) \le cSW(C \cup \{c\}) - cSW(C)$ for any $c, c' \in \mathcal{C}, c \subset \mathcal{C}$. This means that social welfare can be approximately optimized by greedily adding providers until no further viable providers can be added.

A more efficient alternative to greedy provider selection is to directly round the results of the LP relaxation of Eq. (4). This approach has been found to perform well on similar problems (see e.g. Jones (2015)). In preliminary experiments, w that the LP-rounding heuristic performs indistinguishably from the greedy method in terms of social welfare. Given its superior computational performance, we only evaluate the LP rounding approach in our experiments (denoted LP-RS) in Sec. 4. Note the techniques for scaling up submodular maximization and linear programming are relatively well understood (e.g. Mirzasoleiman et al. (2016); Boyd et al. (2011)), hence we will not discuss large-scale implementations within this paper.

The weighted linear utility model provides us with a mechanism for modeling *decreasing marginal returns* in user utility, specifically, by discounting user rewards by setting $\alpha_t = \gamma^{t-1}$ for some $\gamma \in (0, 1]$. Such discounting is one simple way to model the realistic assumption of decreasing marginal utility with increased content consumption (e.g., a good match for a second user query has less impact on her utility than it does for the first query). This model makes it easier to maintain provider viability and improve social welfare with lower individual regret (see Sec. 4).

### 3.2. Non-linear Utility: A Column Generation Model

While additive utility provides us with a useful model class that can be solved approximately in polynomial time, it cannot express important structures such as sigmoidal utility (Kahneman & Tversky, 1979). Optimal matching with such non-linearities can be challenging.

Suppose now that $f$ is nonlinear, e.g., $f(\mathbf{r}^\pi(u)) = \sigma(\mathbf{r}^\pi(u) + \beta)$. The facility location literature provides little guidance for such problems. While approximation results are known for concave problems (Hajiaghayi et al., 2003), it is unclear if these apply with constraints, a setting that, we believe, has not been investigated.

Our approach linearizes the problem to form a MILP that can subsequently be relaxed to an LP as follows. Let $C \in \mathcal{C}^k$ be a $k$-tuple of providers. A pair $(q_u, C) \in \mathcal{Q} \times \mathcal{C}^k$ represents a possible answer to user $u$'s $k$ queries identical to $q_u$ by the provider tuple $C$. We call such a tuple a *star* $q_u C$. For each star, we use a variable $\pi_{q_u C}$ to represent the policy's match to $q_u$. The linearized objective is then:

$$\text{maximize}_{\pi,y} \sum_{u \in \mathcal{U}} \sum_{q_u \in \mathcal{Q}} \sum_{C \in \mathcal{C}^k} \pi_{q_u C} \bar{\sigma}(q_u, C) \ , \quad (5)$$

where $\bar{\sigma}(q_u, C) = \rho(u) P_u(q_u) \sigma(q_u, C)$. The linear constraints from Eq. (4) are adapted to these stars. See Appendix A.2.1 for the complete optimization formulation and our use of column generation to solve this much larger, more complex problem.

### 3.3. Incorporating Regret Trade-offs

As discussed in Sec. 2.2, pure social welfare maximization can sometimes induce large maximum regret (i.e., high regret for some users). Fortunately, max regret can be traded off against social welfare directly in the optimization. Let $\mu_u$ be a constant denoting $u$'s maximum utility $U_{\infty}^{\pi_u^*}(u)$ (see Sec. 2.3). Since the term $U^{\pi}(u) = \sum_{t=1}^{T} \sum_{q_u \in \mathcal{Q}} \sum_{c \in \mathcal{C}} \pi_{q_u,c,t} \bar{r}(q_u, c, t)$ denotes $u$'s expected utility in LP (4), we can express $u$'s realized regret as a variable $Rgrt_u \mu_u - U^{\pi}(u)$ for all $u \in \mathcal{U}$. Letting variable $MR$ represent the max regret induced by the policy, we can add the term $-\lambda MR$ to the objective in MILPs (4) and (5) where constant $\lambda$ controls the desired welfare-regret trade-off. Constraining $MR \geq Rgrt_u, \forall u \in \mathcal{U}$ ensures $MR$ takes on the actual max regret induced by $\pi$.

### 3.4. Extensions of the Model

Our matching optimization relies on some restrictive, unrealistic assumptions about real-world RSs. However, our stylized model directly informs approaches to more realistic models (and can sometimes be adapted directly).

**Robustness and RL:** Strong assumptions about query distribution variance motivated our equilibrium arguments. Even if these assumptions do not hold, our LP formulation can be extended to generate "high-probability" optimal equilibria. For example, lower-confidence bounds on the traffic expected for each provider under the induced policy can be constructed, and viability thresholds inflated to allow for a margin of safety. This can be encoded formally or heuristically within our LP model. A full RL approach offers more flexibility by dynamically adjusting the queries matched to a provider as a function of the state of all providers (e.g., exploiting query variance to opportunistically make additional providers viable, or "salvage" important providers given, say, an unanticipated drop in traffic). The combinatorial nature of the state space (state of engagement of each provider) complicates any RL model. Adaptive online methods (e.g., as used in ad auctions (Mehta, 2013)) can exploit optimization techniques like ours to dynamically adjust the matching without the full complexity of RL.

**Incomplete Information:** Our model assumes complete knowledge of both user utility and "interests" (via the query distribution, the reward function and the utility function), and of a provider's position in "topic space" and its utility (via its viability). In practice, these quantities are learned from data and constantly updated, are generally never known with full precision, and are often very uncertain. Indeed, one reason "unfair" recommendations arise is when the RS does not undertake sufficient exploration to discover diversity in user interests (especially for niche users). Likewise an RS is usually somewhat (though perhaps less) uncertain of a provider's content distribution. Incorporating this uncertainty into our model can (a) generate more robust matchings, and (b) drive exploration strategies that uncover *relevant* user interests and utility given ecosystem viability constraints. These are important directions for future work.

**Richer Dynamics:** More complex dynamics exist in real RS ecosystems than are captured by our simple model, including: arrival/departure of providers/users; evolving user interests and provider topics/quality; richer provider responses (e.g., not just abandonment, but quality reduction, content throttling, topic shifts); and strategic behavior by providers. Our model serves only as a starting point for richer explorations of these phenomena.

## 4. Experiments

We evaluate our LP-rounding method (Sec. 3.1), dubbed LP-RS, for additive utility models (we consider both cumulative and discounted reward). Preliminary experiments show that LP-RS runs faster and performs better than the greedy/submodular approach, so we do not present results for the latter here. We provide a detailed evaluation of column generation for nonlinear models (Sec. 3.2) in Appendix A.2.3. We compare the policy $\pi_{\text{LP}}$ based on LP-RS to a myopic baseline policy[4] $\pi_{\text{My}}$ w.r.t. social welfare, max regret and provider diversity/viability, assessing both on several domains using a RS ecosystem simulator that captures provider viability dynamics. We outline the simulator, describe our datasets and trained embeddings, then discuss our findings. We use the RecSim framework for simulating RS environments (Ie et al., 2019a), and the main experiments can be found in the RecSim codebase.[5]

[4] Appendix D compares to an affinity-aware stochastic policy.

[5] https://github.com/google-research/recsim/blob/master/README.md#Papers

### 4.1. Ecosystem

The ecosystem simulator captures the provider viability dynamics described in Sec. 2.1. At each epoch, the RS observes a single query per user, and must serve a "slate" of $s$ providers to each user. A provider $c$ can only occur *once* in each slate (e.g., the myopic policy rank-orders the $s$ best-fit providers). The user query at each epoch is sampled as $q_u[t] \sim P_{u[t]}(X)$; hence, the realized preference changes at each epoch, but the distribution parameters are fixed throughout the simulation. Providers are static and all viability thresholds have the same value $\nu$.

### 4.2. Datasets

**Synthetic data** To examine the emergent properties of the ecosystem in a controlled setting, we generate synthetic data in a two-dimensional topic space, like those in Fig. 2. The query distribution is a mixture of Gaussians, with one component for each provider centered at that provider's location in topic space $X$, and its weight reflecting the user's affinity for that provider's topic area. We consider two variants of the mixture model: (a) a *uniform* variant where providers/topics are distributed uniformly, and all users have the same variance; and (b) a *skewed* variant, where (w.l.o.g.) topics near the origin are considered *popular* and receive relatively more users, while topics far from the origin are *niche* with fewer users, but whose users are more *loyal* and exhibit lower variance. User-provider reward is given by $f(a, b) = -||a - b||_2$ (with max value 0).

**Movie ratings** We train an embedding on the Movielens dataset (Harper & Konstan, 2015) using non-negative matrix factorization on a sparse matrix of user-movie engagements (see Appendix B for details). The pair of matrix factors are then used as embedding vectors for users and providers for the simulator (here each provider is a movie). User-provider rewards are computed as $r(q_u, c) = q_u^T c$.

**Social network** We train user and provider embeddings from the SNAP 2010 Twitter follower dataset (Kwak et al., 2010). This dataset consists of a large list of (followee, follower) pairs, each with a unique account ID. We designate popular accounts as "providers," and other accounts as "users," then learn a low-dimensional embedding that captures affinity between users and providers via a link prediction task (see Appendix B for details). User-provider rewards are computed as $r(q_u, c) = q_u^T c$.

### 4.3. Results

**Exploring Embedding Type** We begin by using synthetically generated embeddings to study how properties of the *embeddings* affect long-term social welfare under $\pi_{\text{My}}$. We evaluate $\pi_{\text{LP}}$ and $\pi_{\text{My}}$ using the uniform and skewed synthetic embeddings. The results (Table 1) show that when user/provider embeddings are *skewed*, myopic recommendation yields suboptimal user welfare due to less popular providers abandoning the platform. LP-RS improves welfare and increases the number of viable providers in both cases. For the remainder of the paper we use only the skewed variant of the synthetic data.

| Data Type | Method | Avg. Welfare | Viable Providers |
|---|---|---|---|
| Uniform | Myopic | -2.41 ± 0.59 | 43.80 ± 1.94 |
| | LP-RS | **-1.23 ± 0.57** | **48.00 ± 2.10** |
| Skewed | Myopic | -5.68 ± 0.61 | 34.40 ± 1.96 |
| | LP-RS | **-3.40 ± 1.10** | **42.00 ± 3.35** |

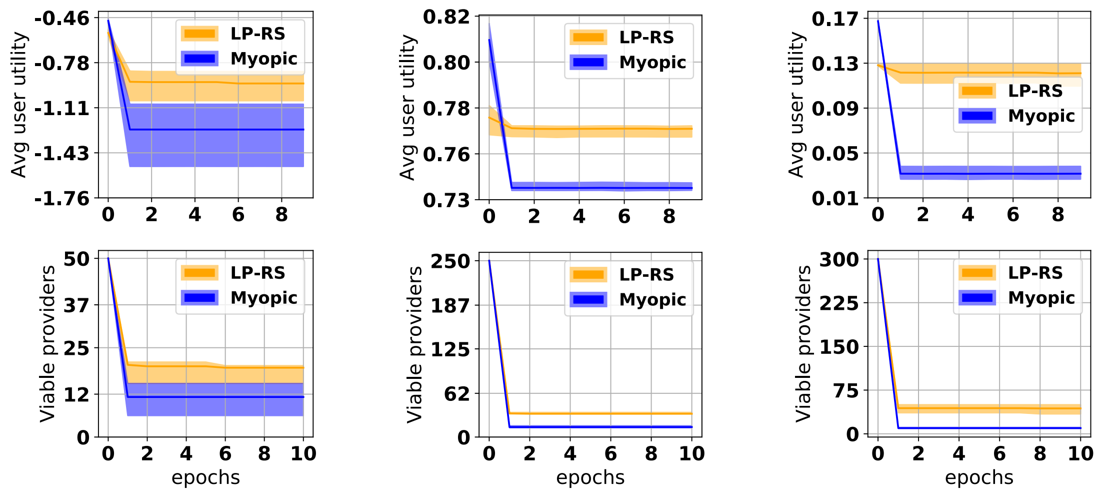
*Table 1.* (Synthetic data) Myopic recommendation performs poorly when observing *skewed* user and provider embeddings (some topic space areas are more popular). LP-RS improves social welfare *and* number of viable providers for both data types. Note that zero is the maximum possible welfare in this setting.

**Tradeoffs in Regret and Welfare** Next we investigate the trade-off between social welfare and individual regret induced by LP-RS at various levels of diminishing returns for user utility, introduced by discounting immediate user rewards as discussed in Sec 3.1. Recall that $Rgrt^{\pi}(u)$ is defined w.r.t. a policy $\pi_u^*$ that is "tailor made" for user $u$, one that keeps all providers with closest affinity to $u$ viable without regard to other users (and will generally serve most users poorly). Under this definition, every policy will generally have very high max regret. But this serves as a useful reference point for understanding how the preferences of any single user trade off with long-term social welfare under a realistic policy $\pi \in \{\pi_{\text{My}}, \pi_{\text{LP}}\}$. We discuss the results for $\pi_{\text{LP}}$ in the following. Results for $\pi_{\text{My}}$ can be found in Appendix F.

We expect that steeper rates of diminishing returns (lower

| $\gamma$ | Avg. Welfare | Max Regret | Regret-Welfare Ratio | Surviving Providers |
|---|---|---|---|---|
| 0.1 | 18.02 ± 1.05 | 7.24 ± 0.77 | 0.40 ± 0.04 | 47.20 ± 1.72 |
| 0.18 | 19.79 ± 1.16 | 7.97 ± 0.84 | 0.40 ± 0.04 | 47.20 ± 1.72 |
| 0.26 | 21.87 ± 1.28 | 8.84 ± 0.91 | 0.41 ± 0.04 | 46.60 ± 2.15 |
| 0.35 | 24.30 ± 1.43 | 10.14 ± 1.18 | 0.42 ± 0.05 | 45.20 ± 3.06 |
| 0.43 | 27.17 ± 1.58 | 11.65 ± 0.91 | 0.43 ± 0.03 | 44.00 ± 3.16 |
| 0.51 | 30.44 ± 1.79 | 14.19 ± 1.46 | 0.47 ± 0.06 | 44.80 ± 0.98 |
| 0.59 | 34.30 ± 1.95 | 16.50 ± 1.73 | 0.48 ± 0.06 | 43.00 ± 1.90 |
| 0.67 | 38.67 ± 2.29 | 20.53 ± 1.46 | 0.53 ± 0.06 | 42.60 ± 1.50 |
| 0.75 | 43.69 ± 2.61 | 24.61 ± 1.31 | 0.57 ± 0.05 | 41.80 ± 1.72 |
| 0.84 | 49.51 ± 2.90 | 28.95 ± 1.11 | 0.59 ± 0.04 | 39.80 ± 3.06 |
| 0.92 | 56.15 ± 3.26 | 33.05 ± 1.16 | 0.59 ± 0.04 | 36.80 ± 3.97 |
| 1.0 | 63.62 ± 3.58 | 37.89 ± 1.39 | 0.60 ± 0.04 | 34.20 ± 5.08 |

*Table 2.* The trade-off between average user welfare and max user regret depends on the discounting factor $\gamma$. The RS makes multiple recommendations to each user during each epoch, and a lower $\gamma$ indicates more steeply discounted returns for recommendations beyond the first one.




(a) Synthetic data embeddings. (b) Movielens data embeddings. (c) SNAP data embeddings.

*Figure 3.* Average social welfare and number of viable providers per epoch of simulation. Bold lines are means over 5 seeds, while shaded regions show 25-th to 75-th percentiles across runs.

$\gamma$) should lead to lower costs—that is, lower individual regret, or sacrifice of individual utility—to generate the optimal provider "subsidies" and generally induce more diverse provider sets in equlibrium, which in turn leads to lower max regret. This makes our social welfare objective more aligned with both provider and user fairness. Table 2 corroborates this intuition, showing that better max-regret-to-average-welfare ratios are generally achieved at low levels of $\gamma$, as are a greater number of viable providers.

**Large-scale Simulations** We carry out large-scale simulations using the dataset embeddings described in Section 4.2 (see Appendix C for details). Fig. 3 shows results from the simulations, tracking both the number of viable providers and average user utility (i.e., social welfare divided by number of users). We find that $\pi_{\text{LP}}$ quickly converges to an equilibrium that sustains more providers than $\pi_{\text{My}}$. Average user utility is also improved under $\pi_{\text{LP}}$. Fig. 4 shows an example user utility histogram (aggregated over the entire simulation). LP-RS has an overall positive impact on the distribution of user utility, relative to the myopic baseline. However the increase in social welfare comes at the cost of decreased utility for some of the most well-off users. See Appendix E for utility histograms for all datasets.

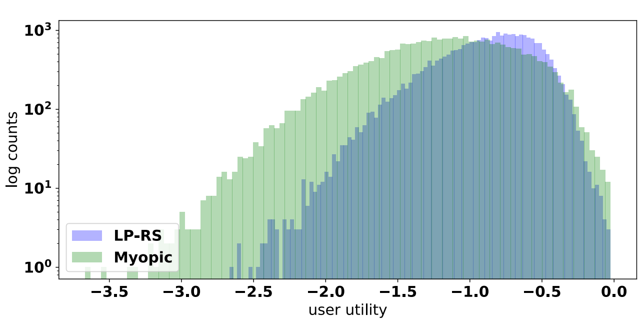



*Figure 4.* User utility histogram (Synthetic embeddings). The LP recommender improves average social welfare at the expense of some maximum user regret.

## 5. Related Work

"Fairness" in outcomes for users and providers could be described in a variety of (possibly conflicting) ways, and any computational measurement of fairness in this context will surely reflect normative principles underpinning the RS design (Binns, 2018; Leben, 2020). We have presented a utilitarian view whereby fair outcomes are realized when the average welfare of users is maximized in equilibrium, without taking other factors such as user/provider demographic group membership or diversity of content into account explicitly. This is consistent with some recent approaches that model long-term fairness by constraining the exploration strategy of the decision maker (Joseph et al., 2016; Jabbari et al., 2017) in the sense that fairness and optimality (here w.r.t. expected user rewards) are aligned. However this approach may not always be appropriate, so we note that the general matching strategy we employ captures the *dynamics* of the RS and could in principle be adapted to encourage other types of "fair" outcomes in accordance with different normative principles.

Studies of fairness in ranking typically consider notions of *group fairness*, for example, by regularizing standard ranking metrics to encourage parity of the ranks across demographic groups (Yang & Stoyanovich, 2016; Zehlike et al., 2017). Celis et al. (2017) propose a matching algorithm for ranking a set of items efficiently under a fairness constraint that encourages demographic diversity of items in the top-ranked position.

Research that considers fairness w.r.t. *content providers* (rather than users), fairly allocating the exposure of providers to users (Singh & Joachims, 2018; Biega et al., 2018), is closely related to our work, especially in some aspects of its motivation. These models impose fairness constraints that address "provider" concerns, and maximize user utility subject to these constraints. In this view, the policy makes commitments to every provider in the ecosystem, while user welfare is secondary to satisfying the fairness constraints. This requires that fairness constraints be crafted very carefully (which is a non-trivial problem (Asudeh et al., 2019)) so as to not have an undue impact on user welfare. In our utilitarian framework, provider fairness is justified by user welfare, implicitly incorporating fairness constraints.

Also very related to our work is the work of Ben-Porat & Tennenholtz (2018), who develop a game-theoretic model of RSs whose providers act strategically—by making available or withholding content—to maximize the user engagement derived from an RS platform. They analyze the policies of the RS: using an axiomatic approach, they prove no RS policy can satisfy certain properties jointly (including a form of fairness); and in a cooperative game model, they show the uniqueness and tractability of a simple allocation policy. While related, their models are not dynamic and do not (directly) assess complex user utility models; but they examine more complex, *strategic* behavior of providers in a way we do not. Ben-Porat et al. (2019) draw a connection between (strategic) facility location games and RSs with strategic providers. Our models relate to non-strategic facility location, with an emphasis on scalable optimization methods.

*Individual fairness* provides an important alternative perspective, requiring that two individuals similar w.r.t. a task should be classified (or otherwise treated) similarly by the ML system (Dwork et al., 2012). Our utilitarian approach guarantees that providers with sufficient impact on user welfare remain viable, regardless of whether their audience lies at the tail or head of content space. As shown in Sec. 4, this provides a dramatic improvement over myopic policies which tend to serve "head" users and providers disproportionately well. While maximizing welfare does not guarantee high individual utility, we generally expect high individual utility to emerge across the user population if utility functions exhibit diminishing returns. If the form of user utility precludes this, the objective can be augmented with a maximum individual regret term as discussed in Sec 3.3.

Also relevant is recent research extending algorithmic fairness to dynamical systems. Several methods for improved fairness in sequential decision-making have been proposed, including work on bandits (Joseph et al., 2016), RL (Jabbari et al., 2017), and importance sampling (Doroudi et al., 2017). Fairness in dynamical systems has been explored in specific domains: predictive policing (Lum & Isaac, 2016; Ensign et al., 2018), hiring (Hu & Chen, 2018; Hu et al., 2019), lending (Mouzannar et al., 2019), and RSs (Chaney et al., 2018; Bountouridis et al., 2019). In general, this work has focused on the role of algorithms in shaping environments over time (Hashimoto et al., 2018; Kannan et al., 2019), observing that the repeated application of algorithms in a changing environment impacts fairness in the long-term differently from short-term effects.

# 6. Conclusion

We have developed a stylized model of the ecosystem dynamics of a content recommender system. We have used it to study the effects of typical myopic RS policies on content providers whose viability depends on attaining a certain level of user engagement. We showed that myopic policies can serve users poorly by driving the system to an equilibrium in which many providers fail to remain viable, inducing poor long-term (user) social welfare. By formulating the recommendation problem holistically as an optimal constrained matching, these deficiencies can be overcome: we optimize long-term social welfare, while at the same time increasing *provider* viability despite the fact that our objective is to increase *user* welfare. We developed several algorithmic approaches to the matching problem and experiments with our LP-based approach showed significant improvements in user welfare over myopic policies. While our model is stylized, we believe it offers insights into more general, realistic RS model as outlined in Sec. 3.4. It provides a rich framework for studying tradeoffs between individual utility and (utilitarian) social welfare. Extensions to account for group fairness, strategic behavior and exploration policies are critical areas of future research as are new algorithmic techniques (e.g., reinforcement learning, online matching) as discussed in Sec.3.4.

# Acknowledgements

Thanks to Francois Belletti, Yi-fan Chen and Aranyak Mehta for valuable discussions on this topic and to the reviewers for their helpful suggestions.

## References


An, H., Singh, M., and Svensson, O. LP-based algorithms for capacitated facility location. *SIAM Journal on Computing*, 46(1):272–306, 1 2017. ISSN 0097-5397. doi: 10.1137/151002320.

Asudeh, A., Jagadishy, H., Stoyanovichz, J., and Das, G. Designing fair ranking schemes. *ACM SIGMOD Record*, 01 2019.

Barocas, S. and Selbst, A. D. Big data's disparate impact. *California Law Review*, 671, 2016.

Ben-Porat, O. and Tennenholtz, M. A game-theoretic approach to recommendation systems with strategic content providers. In *Advances in Neural Information Processing Systems 31 (NeurIPS-18)*, pp. 1118–1128, Montreal, 2018.

Ben-Porat, O., Goren, G., Rosenberg, I., and Tennenholtz, M. From recommendation systems to facility location games. In *Proceedings of the Thirty-third AAAI Conference on Artificial Intelligence (AAAI-19)*, pp. 1772–1779, Honolulu, 2019.

Beutel, A., Covington, P., Jain, S., Xu, C., Li, J., Gatto, V., and Chi, E. H. Latent cross: Making use of context in recurrent recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining (WSDM-18)*, pp. 46–54, Marina Del Rey, CA, 2018.

Beutel, A., Chen, J., Doshi, T., Qian, H., Wei, L., Wu, Y., Heldt, L., Zhao, Z., Hong, L., Chi, E. H., and Goodrow, C. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD-19)*, pp. 2212–2220, Anchorage, AK, 2019.

Biega, A. J., Gummadi, K. P., and Weikum, G. Equity of attention: Amortizing individual fairness in rankings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pp. 405–414, New York, NY, USA, 2018. Association for Computing Machinery.

Binns, R. Fairness in machine learning: Lessons from political philosophy. In *Conference on Fairness, Accountability and Transparency*, pp. 149–159, 2018.

Bountouridis, D., Harambam, J., Makhortykh, M., Marrero, M., Tintarev, N., and Hauff, C. SIREN: A simulation framework for understanding the effects of recommender systems in online news environments. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 150–159. ACM, 2019.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011. ISSN 1935-8237. doi: 10.1561/2200000016. URL `https://doi.org/10.1561/2200000016`.

Celis, L. E., Straszak, D., and Vishnoi, N. K. Ranking with fairness constraints. In *ICALP*, 2017.

Celma, Ò. The long tail in recommender systems. In *Music Recommendation and Discovery*, pp. 87–107. Springer, 2010.

Chaney, A. J. B., Stewart, B. M., and Engelhardt, B. E. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, pp. 224–232, New York, NY, USA, 2018. Association for Computing Machinery.

Chen, M., Beutel, A., Covington, P., Jain, S., Belletti, F., and Chi, E. Top-k off-policy correction for a REINFORCE recommender system. In *12th ACM International Conference on Web Search and Data Mining (WSDM-19)*, pp. 456–464, Melbourne, Australia, 2018.

Cornuejols, G., Fisher, M. L., and Nemhauser, G. L. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science*, 23(8):789–810, 1977. URL `http://www.jstor.org/stable/2630709`.

Covington, P., Adams, J., and Sargin, E. Deep neural networks for YouTube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 191–198, Boston, 2016.

Doroudi, S., Thomas, P. S., and Brunskill, E. Importance sampling for fair policy selection. In *Uncertainty in Artificial Intelligence (UAI)*, 2017.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pp. 214–226. Association for Computing Machinery, 2012.

Ensign, D., Friedler, S. A., Neville, S., Scheidegger, C., and Venkatasubramanian, S. Runaway feedback loops in predictive policing. In *Conference on Fairness, Accountability and Transparency*, pp. 160–171, 2018.

Hajiaghayi, M. T., Mahdian, M., and Mirrokni, V. S. The facility location problem with general cost functions. *Networks*, 42(1):42–47, 2003.

Harper, F. M. and Konstan, J. A. The Movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, 5(4):1–19, 2015.

Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, 2018.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW-17)*, pp. 173–182, Perth, Australia, 2017.

Hu, L. and Chen, Y. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*, pp. 1389–1398. International World Wide Web Conferences Steering Committee, 2018.

Hu, L., Immorlica, N., and Vaughan, J. W. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 259–268. ACM, 2019.

Hu, Y., Koren, Y., and Volinsky, C. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*, pp. 263–272. Ieee, 2008.

Ie, E., Hsu, C.-w., Mladenov, M., Jain, V., Narvekar, S., Wang, J., Wu, R., and Boutilier, C. Recsim: A configurable simulation platform for recommender systems. *arXiv preprint arXiv:1909.04847*, 2019a.

Ie, E., Jain, V., Wang, J., Narvekar, S., Agarwal, R., Wu, R., Cheng, H.-T., Chandra, T., and Boutilier, C. SlateQ: A tractable decomposition for reinforcement learning with recommendation sets. In *Proceedings of the Twenty-eighth International Joint Conference on Artificial Intelligence (IJCAI-19)*, pp. 2592–2599, Macau, 2019b.

Jabbari, S., Joseph, M., Kearns, M., Morgenstern, J., and Roth, A. Fairness in reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1617–1626. JMLR.org, 2017.

Jacobson, K., Murali, V., Newett, E., Whitman, B., and Yon, R. Music personalization at Spotify. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys16)*, pp. 373–373, Boston, Massachusetts, USA, 2016.

Jones, M. The maximum facility location problem. B.s. thesis, The University of Sydney, Sydney, Australia, 2015.

Joseph, M., Kearns, M., Morgenstern, J. H., and Roth, A. Fairness in learning: Classic and contextual bandits. In *Advances in Neural Information Processing Systems*, pp. 325–333, 2016.

Kahneman, D. and Tversky, A. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263–292, 1979.

Kannan, S., Roth, A., and Ziani, J. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 240–248. ACM, 2019.

Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., and Riedl, J. GroupLens: Applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

Kwak, H., Lee, C., Park, H., and Moon, S. What is Twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, pp. 591–600, 2010.

Leben, D. Normative principles for evaluating fairness in machine learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, pp. 86–92, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371100. doi: 10.1145/3375627.3375808. URL https://doi.org/10.1145/3375627.3375808.

Li, S. On facility location with general lower bounds. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '19, pp. 2279–2290, USA, 2019. Society for Industrial and Applied Mathematics.

Lum, K. and Isaac, W. To predict and serve? *Significance*, 13(5):14–19, 2016.

Mehta, A. Online matching and ad allocation. *Foundations and Trends in Theoretical Computer Science*, 8(4):265–368, 2013.

Mirzasoleiman, B., Karbasi, A., Sarkar, R., and Krause, A. Distributed submodular maximization. *Journal of Machine Learning Research*, 17(235):1–44, 2016. URL http://jmlr.org/papers/v17/mirzasoleiman16a.html.

Mladenov, M., Creager, E., Ben-Porat, O., Swersky, K., Zemel, R., and Boutilier, C. Optimizing long-term social welfare in recommender systems: A constrained matching approach. Technical report, 2020. arXiv:2008.00104.

Mouzannar, H., Ohannessian, M. I., and Srebro, N. From fair decision making to social qquality. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 359–368. ACM, 2019.

Ribeiro, M. H., Ottoni, R., West, R., Almeida, V. A. F., and Jr., W. M. Auditing radicalization pathways on YouTube. In *FAT* '20: Conference on Fairness, Accountability, and Transparency*, pp. 131–141, Barcelona, 2020.

Salakhutdinov, R. and Mnih, A. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems 20 (NIPS-07)*, pp. 1257–1264, Vancouver, 2007.

Singh, A. and Joachims, T. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, pp. 2219–2228, New York, NY, USA, 2018. Association for Computing Machinery.

Yang, K. and Stoyanovich, J. Measuring fairness in ranked outputs. In *SSDBM '17*, 2016.

Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., and Baeza-Yates, R. Fa*ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pp. 1569–1578, New York, NY, USA, 2017. Association for Computing Machinery.