Bandit based Optimization of Multiple Objectives on a Music Streaming Platform

Rishabh Mehrotra[†]
Spotify, London
rishabhm@spotify.com

Niannan Xue[†*]
Imperial College London
n.xue15@imperial.ac.uk

Mounia Lalmas
Spotify, London
mounia@acm.org

ABSTRACT

Recommender systems powering online multi-stakeholder platforms often face the challenge of jointly optimizing for multiple objectives, in an attempt to efficiently match suppliers and consumers. Examples of such objectives include user behavioral metrics (e.g. clicks, streams, dwell time, etc), supplier exposure objectives (e.g. diversity) and platform centric objectives (e.g. promotions). Jointly optimizing multiple metrics in online recommender systems remains a challenging task. Recent work has demonstrated the prowess of contextual bandits in powering recommendation systems to serve recommendation of interest to users. This paper aims at extending contextual bandits to multi-objective setting so as to power recommendations in a multi-stakeholder platforms.

Specifically, in a contextual bandit setting, we learn a recommendation policy that can optimize multiple objectives simultaneously in a fair way. This multi-objective online optimization problem is formalized by using the Generalized Gini index (GGI) aggregation function, which combines and balances multiple objectives together. We propose an online gradient ascent learning algorithm to maximise the long-term vectorial rewards for different objectives scalarised using the GGI function. Through extensive experiments on simulated data and large scale music recommendation data from Spotify, a streaming platform, we show that the proposed algorithm learns a superior policy among the disparate objectives compared with other state-of-the-art approaches.

ACM Reference Format:
Rishabh Mehrotra[†], Niannan Xue[†*], and Mounia Lalmas. 2020. Bandit based Optimization of Multiple Objectives on a Music Streaming Platform. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3394486.3403374

1 INTRODUCTION

Platform ecosystems have witnessed an explosive growth by facilitating efficient interactions between multiple stakeholders, including e.g. buyers and retailers (Amazon), guests and hosts (AirBnb), riders and drivers (Uber), and listeners and artists (Spotify). A large number of such platforms rely on machine learning powered matching engines connecting consumers with suppliers by acting as a central platform, thereby finding the right fit and efficiently mediating economic transactions between the two sides.

Recent advancements in understanding, interpreting and leveraging user behavioral signals have enabled such recommender systems to be optimized for many different user centric objectives, including clicks [14], dwell time [43], session length time [13], streaming time, conversion [32] among others. Beyond user-centric objectives, platforms can additionally optimize their models for different stakeholder objectives, including exposure, fairness, diversity, promotion and revenue metrics.

For example, a recommender system powering a multi-stakeholder platform (e.g. Amazon, Uber) would not only aim at serving recommendations maximising user satisfaction, but also would optimise for fair exposure to retailers and suppliers, alongside maximising revenue-related objectives [25].Even in the case of user-centric recommendation systems, predicting user interest alone is not enough to ensure user satisfaction, and notions such as engagement, novelty and diversity greatly enhance user experiences [8] and should be promoted as well. This motivates the need for developing multi-objective recommendation models capable of leveraging multiple user-centric as well as other stakeholder objectives.

In this work we consider the task of designing multi-objective recommender systems for online multi-stakeholder platforms. Specifically, we propose a multi-objective formulation of a contextual bandit based recommender system. Multi-armed bandits (MAB) and their variants (contextual bandits, dueling bandits, etc.) are increasingly popular in sequential decision making scenarios, wherein the system faces a dilemma over whether to explore to discover more about user preferences, or to exploit current knowledge and serve recommendations that pique user interests. In the contextual MAB, different from the classical MAB, the system observes a context (side information) at the beginning of each round, which gives a hint about the expected arm rewards in that round. Such contextual bandit based models are better adept at handling uncertainty of relevance of items and new information (new users, new items) than traditional recommender systems.

In traditional bandit settings, only a single scalar feedback is observed after an action is performed. However, multi-objective systems necessarily involve joint optimisation of a number of different criteria simultaneously. While a lot of research has gone in developing contextual bandit solutions [18, 33], multi-objective variants of contextual bandits have received relatively less attention.

In this paper, we focus on developing multi-objective contextual bandit models for digital multi-stakeholder platforms. We focus

[†]Equal contribution authors.
[*] This work was done while the author was an intern at Spotify.

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403374

on the contextual bandit problem where the reward is assumed to be a noisy linear function of context, i.e. side information, not for just a single objective, but for a number of objectives. We propose a multi-objective contextual bandit model based on Generalized Gini Index (GGI) [5] by introducing a model that assimilates contextual information and optimizes for a number of different, including competing objectives. Such an extension of bandits is non-trivial because there is no longer a fixed mean feedback for any specific arm. The goal of the proposed GGI based optimization is to be both efficient, i.e., minimize the cumulative cost for each objective, and fair, i.e., balance the different objectives. We propose an online gradient ascent method to maximise a scalarisation of the accumulated reward over several objectives.

We consider the case of Spotify - an online music streaming platform, and define user-centric satisfaction objectives as well as supplier-centric diversity and promotional objectives. We present correlation analysis across these different user- and supplier-centric objectives to motivate the need for multi-objective modeling. We perform large scale experiments on simulated data as well as real world user interaction data, and demonstrate that our proposed algorithm performs better than a number of established baselines and state-of-the-art multi-objective models.

Experiments with multiple user-centric objectives suggest that optimizing for multiple interaction metrics performs better for each metric than optimizing single metric. Beyond multiple user-centric objectives, experiments around adding promotional objectives (e.g. gender promotion) demonstrate that the platform can obtain gain in such promotional-centric or supplier-centric objectives without severe loss in user centric objectives. Our findings have implications on the design of recommender systems powering online digital multi-stakeholder platforms and motivates future work, which we briefly discuss at the end of the paper.

2 BACKGROUND AND RELATED WORK

Multi-objective Optimization. Multi-objective optimisation (aka MOO) is a well studied subject in operation research and machine learning, with applications in multi-agent systems [44], reinforcement learning [10] and, recently, multi-armed bandits [5]. Traditional MOO approaches aimed at seeking the Pareto front directly. A more practical approach involes converting the vectorial objectives into a single one using some aggregation function, a process known as scalarisation [21]. Popular aggregation functions include the weighted sum [15] and the regularized maximin fairness policy [44]. We use a concrete example of the generalised Gini function in this paper, which can be seen as a generalisation to both total ordering and linear priorities [4]. Furthermore, unlike past approaches, which dealt with MOO in traditional supervised learning setup, the focus on this work is on more interactive, online, adaptive learning setting based on user feedback.

MOO in Search & Recommender Systems. MOO has been very effective for various business motivated applications in search and recommender systems, including click shaping [2, 3], email volume optimization [11] and serving recommendations with capacity constraints [6]. Earlier efforts also included extensions of collaborative filtering for MOO [12] and multi-criterion user modeling [17]. In search and information retrieval, prior work has leveraged and developed MOO techniques for optimizing novelty and diversity [30] and learning to rank [35]. Recent approaches on fairness in recommendation have also considered multi-objective models [40]. A recent tutorial also covered such applications in detail [24].

A key difference between our approach and aforementioned MOO approaches is our explicit focus on bandit based settings, which perform better than traditional recommendation approaches at handling uncertainty of item relevance and new information. Our proposed model builds on top of such bandit based approaches and extends the contextual bandits to a multi-objective setting.

Bandits & Multi-objective Bandits. For multi-armed bandits, there has been extensive work to address the multi-objective nature. A comparison of different algorithms using multiple regret evaluation metrics can be found in [9]. Knowledge gradient [29] has been used to explore the Pareto optimal arms. Algorithms such as Hierarchical optimistic optimisation strategy [26], Thompson sampling [42], and combination of bi-objective optimization with combinatorial bandits [38]. have also been proposed. While these approaches tackle the MOO problem in vanilla bandit setting, we focus on the case of contextual bandits, wherein an additional context vector is observed at each iteration, which encodes contextual information about users and content. The proposed method extends multi-objective bandit models to a contextual bandit setting.

While a lot of research has gone in developing multi-objective multi-armed bandits, multi-objective variants of contextual bandits have received less attention. Only algorithms in the restrictive settings of similarity information [37] and a dominant objective [36] exist, and rely on the assumption of access to distances between context-arm pairs to the distances between expected rewards of these pairs. This assumption is fairly restrictive in typical industrial settings, where rewards are derived based on user behavioral data.

3 OBJECTIVES & STAKEHOLDERS

Machine learning systems powering modern recommender systems are optimized for and evaluated upon an increasing number of objectives and metrics. For example, user centric recommender systems such as e-commerce portals optimize for different proxies of user satisfaction, including clicks, dwell time and conversion; whereas multi-stakeholder platforms optimize metrics for its various stakeholders, including guests/hosts (Airbnb), buyers/sellers (Amazon) and listeners/artists (Spotify).

In this paper, we discuss scenarios wherein the system needs to jointly optimize for multiple metrics and propose a bandit model as a solution. In this section, we begin by motivating an industrial use-case of multi-objective modelling and present data driven analysis that motivate the need for multi-objective modelling for recommender systems. We then briefly describe the contextual bandit formulation of the recommender problem, which serves as the base model on top of which the proposed model is built.

Data Context. We consider the specific use case of a global music streaming platform where users listen to music from different artists. The recommendation system recommends a set of tracks (i.e. playlists containing songs) to the user, each of which could come from different artists. Different sets have varying degree of

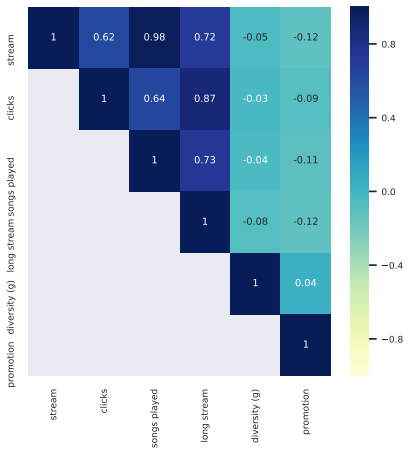

Figure 1: Heatmap of correlations between different objectives.

relevance to user's interests, and users could be satisfied with the recommended set to varying extent.

3.1 Objective Definitions

Often in user centric systems, system designers have access to multiple implicit signals from the rich fine-grained user interaction information logged in behavior logs, which give rise to a number of user-centric objectives. For example, in a case of music streaming services, a system could optimize for clicks, streams, number of songs played or other user engagement metrics. Often, such metrics are correlated, and optimizing for one would inherently lead the model to improve other correlated metrics. However, this needs not be true: an objective might be un-correlated or negatively correlated with user satisfaction metrics, and there exist strict trade-off in optimizing one against the other. For example, recent work has demonstrated that optimizing for relevance hurts diversity [25] and vice-versa, thereby motivating the need for development of multi-objective models that optimize multiple metrics.

Specifically, for the case of music streaming platforms, system designers can optimize for a number of user centric objectives, such as clicks, track stream, number of tracks played, long duration streams, among others. Furthermore, when considering other stakeholders in a music streaming platform, additional objectives surface, including diversity centric and promotion centric objectives.

We conjecture that optimizing for some metrics might hurt other metrics. For example, promoting certain artists in a recommendation setting might annoy users whose taste profiles do not match the artist, and hence hurt user satisfaction metrics. To better understand this interplay between different objectives, we consider a random sample of user streaming data, and estimate and analyze different user centric and artist centric objectives. We consider three user centric objectives: duration of streamed songs (*stream*), clicks on playlists (*clicks*) and number of songs played. Additionally, we consider two artist and platform centric objectives. First, *diversity(g)* quantifies the gender diversity present in the recommended set, which is computed as the percentage tracks in a set whose main artist is a non-male artist. Second, promotion objective assumes a scenario wherein the platform intends to promote some niche artists to users via its recommendations. These diversity and promotional objectives find overlap with other corresponding metrics in other platforms, e.g. diversity and promotion of retailers (e-commerce platforms), funding campaigns (crowdfunding platforms), hosts (AirBnb), among others.

Figure 1 shows the heatmap of the correlation across different objectives. We observe that different objectives are correlated to different extent, with user centric objectives strongly positively correlated, while gender-diversity and promotion objectives are weakly negatively correlated with user centric objectives. This analysis highlights the fact that optimizing for some objectives might have a detrimental effect on other objectives. In cases where the relationship between metrics of different stakeholders are simple and correlated, optimizing one would result in gains in the other metric. But more often than not, balancing in multi-stakeholder platform entails a subtle trade-off between objectives. A recommender system built for optimizing a single metric is ill-suited in a multi-metric multi-stakeholder platform setting, as discussed next.

Before delving into a multiple metric optimization, we first describe a recommendation approach based on contextual bandits, which is widely used to optimize a single user satisfaction metric.

3.2 Contextual Bandit based Recommenders

We formalize the recommendation problem as a combinatorial contextual bandit problem, wherein the recommender system repeatedly interacts with consumers as follows:

(1) the system observes a context ($\mathcal{J}$);
(2) based on the context, the system chooses an action $a \in A$, from the space of K possible actions (sets to recommend);
(3) given a context and an action, a reward $x \in [0, 1]$ is drawn from distribution $D(x|, \mathcal{J})$, with rewards in different rounds being independent, conditioned on contexts and actions.

While the context space can be infinite, composed of information the system has about user's interests, item features and other features like time, location, the action space is finite. We next describe the context and actions used for the presented research.

Context: We leveraged a large number of context signals in the recommendation problem, including: (i) features of the user, such as age range, gender, location, affinity to genres; (ii) features of the playlist such as its artist, its (micro and macro) genres, diversity of songs, popularity; (iii) affinity between the user and the playlist, taking into account past interactions, such as streams, skips, likes, and saves; and (iv) other contextual information, such as the day of the week and the time of day.

Actions: Each action is composed of selecting a set to recommend to the user. In our case of music streaming, we assume a set based recommendation strategy with the user presented with a playlist (a collection of tracks), with each track coming from specific artist.

Rewards: In a traditional user-centric system, the observed reward will be based on how happy the user was with the recommendation served, and the goal of the model is to learn an arm selection strategy that maximizes user satisfaction. Such an arm selection strategy is focused on a single metric, one that is generally chosen as a proxy of user satisfaction. On the other hand, in a multi-stakeholder recommender system, vectorial rewards are observed, one corresponding to each objective, and the arm selection strategy

would be decided based on the strategy that optimises for each of these objectives. We develop such an approach in this paper.

Multi-objective optimization gives us a mathematically principled solution for the trade-off among (often competing) objectives. To this end, we next propose a multi-objective contextual bandit to solve this specific problem.

4 MULTI-OBJECTIVE CONTEXTUAL BANDITS

To jointly optimize multiple objectives for recommending content, we propose a novel multi-objective contextual bandit algorithm. We begin by defining mathematical notations (Section 4.1) and detail the problem set-up (Section 4.2). Section 4.3 describes the arm-selection strategy given multiple objectives and Section 4.4 describes the complete algorithm.

4.1 Notations

Plain letters denote scalars; bold letters denote vectors and calligraphic letters denote matrices, unless otherwise stated. $\boldsymbol{a}_i$ represents the i^{th} element of $\boldsymbol{a}$ and $\mathcal{A}_j$ represents the j^{th} column of $\mathcal{A}$. $\wedge$ is the logical conjunction operator between two conditions. $\mathcal{I}_{m\times m}$ denotes the m by m identity matrix. Projection onto support set $\mathbb{C}$ is given by $\boldsymbol{\Pi}_{\mathbb{C}}$. $\|\boldsymbol{a}\|$ denotes the l_2-norm of vector $\boldsymbol{a}$. For a positive definite matrix $\mathcal{A} \in \mathbb{R}^{d\times d}$, the weighted l_2-norm of vector $\boldsymbol{x} \in \mathbb{R}^d$ is defined by $\|\boldsymbol{x}\|_{\mathcal{A}} = \sqrt{\boldsymbol{x}^T\mathcal{A}\boldsymbol{x}}$. $|a|$ is the absolute value of a.

4.2 Problem Setup

We cast the multi-objective recommendation problem in terms of a multi-arm contextual bandit setting. Assume that we play the bandit problem for a total of T rounds, where each round corresponds to a user session wherein a playlist is recommended to the user (i.e., one bandit arm is selected). For each bandit instance at round t, we are given features $\mathcal{J}_{[t]} = (\mathcal{J}_{[t],1}, \cdots, \mathcal{J}_{[t],K})$ associated with the K possible arms, where $\mathcal{J}_{[t],i} \in \mathbb{R}^M$ and M is the feature length. Such features encode the current user specific context, and may include features representing user taste profiles, historic interaction features and other contextual signals. An arm selection strategy corresponds to selecting a playlist to show to the user given observations about the contextual features in the session. Under the linear shared model, if arm k is chosen at round t, we observe reward

$$\boldsymbol{x}_{[t]} = \mathcal{J}_{[t],k}^T\vartheta^* + \boldsymbol{\zeta}_{[t]}, \tag{1}$$

where $\vartheta^* \in \mathbb{R}^{M\times D}$ is a fixed unknown universal parameter and $\boldsymbol{\zeta}_{[t]} \in \mathbb{R}^D$ is an independent random noise for each objective.

4.3 Arm Selection Strategy

A strategy is a way to pick an arm at each round by examining the features for all arms. This strategy defines which playlist (from a collection of candidate playlists) is shown to the user for each session. We can then calculate a strategy's average reward after T rounds, as

$$\bar{\boldsymbol{x}}_{[T]} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{x}_{[t]}. \tag{2}$$

Algorithm 1 MO-LinCB

Input: K arms, D objectives, T rounds, aggregation function $G(\boldsymbol{x})$, regularisation parameter λ, learning rate η, gradient ascent iterations I
Set $\mathcal{A} = \lambda\mathcal{I}_{M\times M}$
Set $\mathcal{B}_d = \boldsymbol{0}$, $\forall d \le D$
for $t = 1, 2, 3, \ldots, T$ **do**
  Observe K features, $\mathcal{J}_{[t],1}, \mathcal{J}_{[t],2}, \cdots, \mathcal{J}_{[t],K} \in \mathbb{R}^M$
  for $d = 1, 2, 3, \cdots, D$ **do**
    $\hat{\vartheta}_{[t],d} = \mathcal{A}^{-1}\mathcal{B}_d$
  for $k = 1, 2, 3, \cdots, K$ **do**
    $\boldsymbol{\mu}_{[k]} = \mathcal{J}_{[t],k}^T\hat{\vartheta}_{[t]}$
  $\boldsymbol{\alpha}_{[t]} = (1/K, \cdots, 1/K)$
  for $i = 1, 2, 3, \cdots, I$ **do**
    $\boldsymbol{\alpha}_{[t]} = \boldsymbol{\Pi}_{\mathbb{A}}(\boldsymbol{\alpha}_{[t]} + \eta\nabla f_{[t]}(\boldsymbol{\alpha}_{[t]}))$
  Choose arm k according to $\boldsymbol{\alpha}_{[t]}$ and observe reward $\boldsymbol{x}_{[t]}$
  $\mathcal{A} = \mathcal{A} + \mathcal{J}_{[t],k}\mathcal{J}_{[t],k}^T$
  for $d = 1, 2, 3, \ldots, D$ **do**
    $\mathcal{B}_d = \mathcal{B}_d + \boldsymbol{x}_{[t],d} \times \mathcal{J}_{[t],k}$

We follow the scalarisation approach to multi-obejctive optimisation, where one usually wants to compute the Pareto front, or search for a particular element of the Pareto front. In practice, it may be costly (and even infeasible depending on the size of the solution space) to determine all solutions of the Pareto front. One may then prefer to directly aim for a particular solution in the Pareto front. This problem is formalized as a single objective optimization problem, using an *aggregation function*. We employ a Gini index based aggregation function described in detail in Section 4.3.2.

4.3.1 ***Optimal Policy.*** For an aggregation function $G(\boldsymbol{x})$, we aim to seek a strategy such that $G\left(\bar{\boldsymbol{x}}_{[T]}\right)$ is as large as possible, i.e., arms are selected which maximize the aggregation function.

Rather than considering a strategy such that only a single arm is decided at each round, we look for a strategy that, at each round, proposes a probability distribution, $\mathbb{A} = \{\boldsymbol{\alpha} \in \mathbb{R}^K | \sum_{k=1}^K \boldsymbol{\alpha}_k = 1 \wedge 0 \le \boldsymbol{\alpha}_k, \forall k \le K\}$, according to which an arm (i.e. α_k) is to be drawn. That is, we consider **mixed strategies**. For example, we can find an optimal mixed strategy arm selection policy for a single bandit problem with known mean feedback by solving the following optimisation problem,

$$\boldsymbol{\alpha}^* \in \arg\max_{\boldsymbol{\alpha}\in\mathbb{A}} G\left(\sum_{k=1}^{K}\boldsymbol{\alpha}_k\boldsymbol{\mu}_{[k]}\right). \tag{3}$$

In other words, an arm with the highest mean reward is pulled most frequently. Nonetheless, arms with less reward values are also pulled sometimes. This allows the model to trade-off exploitation of known arms with exploration of potentially useful arms.

In the single objective case, arms are compared in terms of their means, which induce a total ordering over arms. In the multi-objective setting, we use a specific form of aggregation criterion to compare arms, which we describe next.

4.3.2 ***Generalised Gini Function.*** The aggregation function allows the model to scalarize inputs from different objectives. We consider a specific example for it, the generalised Gini function

(GGF) [39]. GGF is a non-linear but concave function. It is a special case of the ordered weighted averaging (OWA) aggregation operators [41], which preserves impartiality with respect to the individual criterion. For a reward vector $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_D)$, GGF is defined as

$$G_{\boldsymbol{w}}(\boldsymbol{x}) = \sum_{d=1}^{D} \boldsymbol{w}_d (\boldsymbol{x}_\sigma)_d = \boldsymbol{w}^T \boldsymbol{x}_\sigma, \tag{4}$$

where $\boldsymbol{w}_1 > \boldsymbol{w}_2 > \cdots > \boldsymbol{w}_d > 0$ and σ permutes the elements of $\boldsymbol{x}$ such that $(\boldsymbol{x}_\sigma)_i \leq (\boldsymbol{x}_\sigma)_{i+1}$. GGF is strictly monotonic, which means that a vector that maximises Equation 4 also lies on the Pareto front for direct optimisation of the multiple criteria and different weights ($\boldsymbol{w}$) correspond to different points on the frontier [28]. GGF exhibits a fairness property under the Pigou-Dalton transfer: if $\boldsymbol{x}_i < \boldsymbol{x}_j$, then $G_{\boldsymbol{w}}(\boldsymbol{x}') > G_{\boldsymbol{w}}(\boldsymbol{x})$ for $\boldsymbol{x}'_i = \boldsymbol{x}_i + \epsilon$ and $\boldsymbol{x}'_j = \boldsymbol{x}_j - \epsilon$ where $\epsilon < \boldsymbol{x}_j - \boldsymbol{x}_i$ and $\boldsymbol{x}'_k = \boldsymbol{x}_k$ for $k \neq i, j$. In other words, an equitable transfer of an arbitrarily small amount from a larger component to a smaller component is always preferable. The effect of such a transfer is to balance a reward vector.

Given the GGI formulation of the aggregation function, we next define regret for the multi-objective variant of our bandit model.

4.3.3 ***Regret.*** If we know ϑ^*, then after T rounds, the optimal mixed policy $\boldsymbol{\alpha}^*_{[t]}$ is provided by a solution to the following problem

$$\max_{\boldsymbol{\alpha}_{[t]} \in \mathbb{A}} G\left(\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \vartheta^*\right), \tag{5}$$

where we have assumed that random noises $\boldsymbol{\zeta}_{[t]}$ average out at zero for large T.

We define *regret* as the difference between the optimal value of reward and reward from any strategy as:

$$R_{[T]} = G\left(\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \boldsymbol{\alpha}^*_{[t],k} \mathcal{J}^T_{[t],k} \vartheta^*\right) - G\left(\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \vartheta^*\right), \tag{6}$$

where $\boldsymbol{\alpha}_{[t]}$ is the action recommended by the employed strategy. Note that in our definition we use the true parameter ϑ^* instead of the true mean feedback as in [5]. Also important is to note that performance is measured by the function value of the average reward instead of the average of the rewards' function value.

The arm selection strategy presented above employs the GGF as an aggregation function to scalarize multiple metrics. Our goal now is to find the parameters of the arm selection strategy given by Equation 5. We next describe a gradient ascent based algorithm that learns this policy.

4.4 Learning Algorithm

To surface recommendations that satisfy multiple objectives, we need to learn the arm selection strategy ($\boldsymbol{\alpha}_{[t]}$) that recommends appropriate content that minimizes the regret for an aggregation function $G(\boldsymbol{x})$. We propose an online learning algorithm (**MO-LinCB**) to optimize the regret defined in the previous section. Our method exploits the convexity of the GGI operator and formalizes the policy search problem as an online convex optimization problem, which is solved by Online Gradient Ascent algorithm.

4.4.1 Ridge Regression. The optimal policy that maximizes Equation 5 assumes knowledge of ϑ^*. In this section, we first estimate parameter ϑ^* via l_2-regularised least-squares regression with regularisation parameter $\lambda > 0$ at each round t,:

$$\hat{\vartheta}_{[t]} = \left(\mathcal{J}_{1:t-1}\mathcal{J}^T_{1:t-1} + \lambda \mathcal{I}_{M\times M}\right)^{-1} \mathcal{J}_{1:t-1}\mathcal{X}_{1:t-1}, \tag{7}$$

where $\mathcal{J}_{1:t-1}$ has columns $\boldsymbol{f}_{[1]}, \boldsymbol{f}_{[2]}, \ldots, \boldsymbol{f}_{[t-1]}, \boldsymbol{f}_{[t]}$ being the feature associated with the chosen arm at round t, and $\mathcal{X}_{1:t-1}$ has rows $\boldsymbol{x}_{[1]}, \boldsymbol{x}_{[2]}, \ldots, \boldsymbol{x}_{[t-1]}$. However, if the feature length is huge, matrix inversion is prohibitively challenging and we have to use stochastic gradient descent [31] for ridge regression.[1]

4.4.2 Online Gradient Ascent. We are seeking an algorithm for $\boldsymbol{\alpha}_{[t]}$ such that the objective below is maximised:

$$G\left(\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]}\right). \tag{8}$$

Due to the concavity of $G(\boldsymbol{x})$, we have the following relationship

$$\frac{1}{T}\sum_{t=1}^{T} G\left(\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]}\right) \leq G\left(\frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]}\right). \tag{9}$$

Instead of optimising Equation 8 directly, we use online gradient ascent to optimise the left-hand side of Equation 9. This is because very rarely we have features of all the rounds simultaneously. In the case of personalised recommendation, users always arrive sequentially and an instantaneous response is demanded. Therefore, at each round, we try to optimise

$$G\left(\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]}\right) \tag{10}$$

which can be thought as a function of $\boldsymbol{\alpha}_{[t]}$, $f_{[t]}(\boldsymbol{\alpha}_{[t]})$. To carry out gradient ascent, we take many steps along the gradient of $f_{[t]}(\boldsymbol{\alpha}_{[t]})$. Lastly, we need to project back onto $\mathbb{A}$ to ensure feasibility. The overall algorithm is presented in Algorithm 1.

When using GGF as the aggregation function, Equation 3 can be solved by a linear program [28], so one could also try to optimise Equation 9 by linear programming. However, in live production this is too expensive to deploy due to its prohibitive computational cost. Now the gradient of $f_{[t]}(\boldsymbol{\alpha}_{[t]})$ has an analytical form[2]

$$\frac{\partial f_{[t]}}{\partial \boldsymbol{\alpha}_{[t],k}} = \boldsymbol{w}^T (\mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]})_\sigma$$

and can be used to derive theoretical guarantees. We leave this derivation of guarantees as future work.

The main component in the proposed approach is the application of online gradient ascent on the estimated GGF for the current round, $f_{[t]}(\boldsymbol{\alpha}_{[t]})$. While past approaches in non-contextual version of bandit setting [5] tackle a similar problem, there are differences.

[1] The convergence for a single pass through the data is shown in [27].

[2] Note that σ sorts the components of $\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}^T_{[t],k} \hat{\vartheta}_{[t]}$ in increasing order.

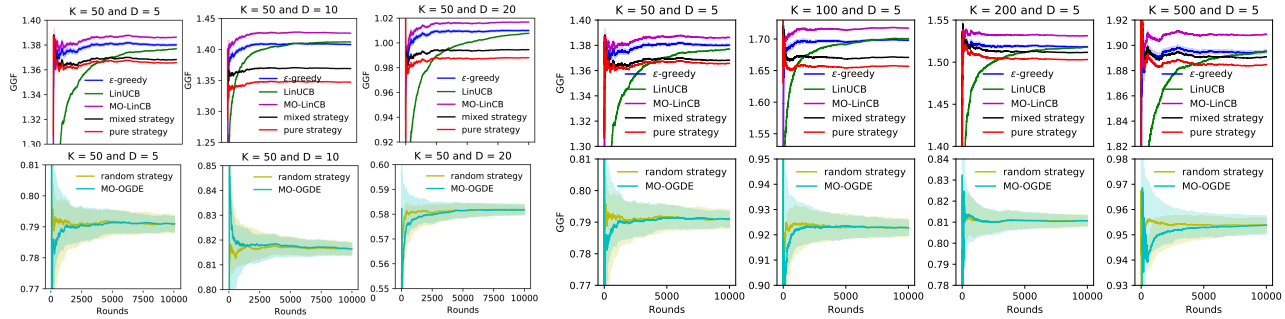

Figure 2: [Simulation test results] GGF of average reward for different number of objectives (D), and for varying number of arms (K). To improve visual inspection of results, we split the results into two parts: top row shows results for 5 compared approaches, while bottom row for the remaining 2 baselines.

First, in the contextual bandit setting, each problem at round t is different from another due to their different mean feedback. Thus, it is not sensible to initialise $\boldsymbol{\alpha}_{[t]}$ for the current round as the outcome from the previous round, $\boldsymbol{\alpha}_{[t-1]}$. To counterbalance such cold start, we iterate multiple times at each round. Second, there is no forced exploration in our algorithm as the means are estimated via ridge regression eliminating the need to pull every arm regularly.

5 EXPERIMENTAL EVALUATION

The proposed multi-objective contextual bandit model allows us to jointly optimize multiple objectives when serving recommendations. To evaluate its performance, we conduct experiments on synthetic dataset and a large scale real world user interaction data from a major music streaming platform. We describe the experimental setup, and present a number of insights that demonstrate that the proposed multi-objective contextual bandits are well suited as recommendation models in multi-stakeholder setting.

5.1 Baseline Methods

We compare the proposed approach (MO-LinCB) with several established methods, including both state-of-the-art bandit recommendation techniques, as well as recent multi-objective bandit algorithms.

(1) ϵ-greedy algorithm (ϵ-g (C), ϵ-g (S), ϵ-g (L)) [34]: a simple bandit approach that instead of picking the best available option always, randomly explores other options with a probability ϵ. The baselines *ϵ-g (C)*, *ϵ-g (S)*, ϵ-g (S) correspond to ϵ-greedy methods for the click, stream length and total number of songs played objectives respectively.
(2) LinUCB [19]: a widely used and deployed bandit approach based on the principle of optimism in the face of uncertainty. It chooses actions by their mean payoffs and uncertainty estimates.
(3) MO-OGDE [5]: a recently proposed multi-objective bandit model that exploits the concavity of GGF and uses forced exploration. The model is non-contextual.
(4) Pure strategy: A baseline model that always pulls the arm with the highest current mean reward.
(5) Mixed Strategy: A baseline model extending the pure strategy baseline, that, at each round, proposes a probability distribution according to which an arm is to be drawn. This is different to the pure strategy baseline mentioned above, wherein only a single arm is decided at each round based on the highest mean reward.
(6) MO-epsilon (*ϵ-g (MO)*): The multi-objective extension of the widely used ϵ-greedy bandit model which uses GGI function.

We additionally consider variants of our proposed approach (MO-LinCB) which is a multi-objective contextual bandit based on GGI, trained with different user satisfactionand supplier diversity objectives:

(1) MO-LinCB: The proposed multi-objective bandit approach that leverages Gini function for equitable distribution across different metrics.
(2) MO-LinCB(g): Variant of the proposed approach, with gender as an additional metric to optimize.

Appendix B provides implementation details of different baselines.

5.2 Datasets

We consider two datasets for evaluation: (i) synthetic data for controlled study of various algorithms and parameters, and (ii) large scale user interaction data from a major music streaming platform.

5.2.1 **Simulated Dataset.** Experimenting with simulated data allows us to vary the number of objectives (D) as well as the number of arms (K), and investigate how the different models would perform for recommendation problems of varying characteristics. We generate random multi-objective contextual bandit problems, with simulated features and rewards. The number of arms K is set to $\{50, 100, 200, 500\}$ and the number of objectives is taken from $D \in \{5, 10, 20\}$. Appendix A provides specific details of the exact simulation setup details.

Evaluation Setup: Each of the competing algorithms, i.e. MO-LinCB, LinUCB, ϵ-greedy, MO-OGDE and random selection is repeated 100 times. As the ground-truth (Equation 5) is too costly to compute, we instead compare with

$$\max_{\boldsymbol{\alpha}_{[t]} \in \mathbb{A}} \frac{1}{T} \sum_{t=1}^{T} G_{\boldsymbol{w}} \left(\sum_{k=1}^{K} \boldsymbol{\alpha}_{[t],k} \mathcal{J}_{[t],k}^{T} \vartheta^{*} \right) \tag{11}$$

on grounds of concavity and its pure counterpart $\max_k G_{\boldsymbol{w}} \left(\mathcal{J}_{[t],k}^{T} \vartheta^{*} \right)$ for each round t. We set the exploration probability of the ϵ-greedy algorithm to 1%. For LinUCB and MO-OGDE, δ is set to 10%. The step size η is set to 5 and the number of iterations I is set to 5 for

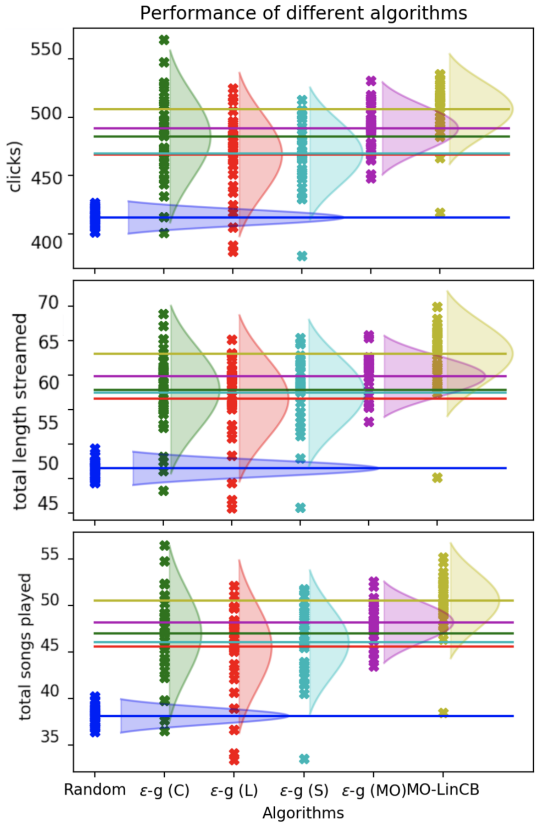

Figure 3: [Music data results] Comparison of different approaches on different online metrics: clicks, total length streamed and total songs played. The baseline *eb* corresponds to the Bart [23] - a contextual bandit recommender system, while *ei, el, es* correspond to ϵ-greedy methods for the click, stream length and total number of songs played objectives respectively.

MO-LinCB. We use this dataset in Sections 5.3 and 5.7 to compare performance of approaches on reward obtained and time complexity based scalability test, respectively.

5.2.2 ***Music Streaming Dataset***. We additionally perform experiments on music streaming data of over 10M users and 12M playlists from a major music streaming platform. The dataset is composed of logged feedback data from live traffic from a random sample of users in a 7 day period, spanning over 200K user interactions with 12 million playlists, spread over 1 million impressions. For playlists that get impressed to users, we record three user metrics, (i) *clicks* (C), (ii) *total length of music streamed* (L) and (iii) *total number of songs played* (S).

The data is split into two buckets: the first bucket contains 30, 000 requests and is used to tune the parameters of each competing algorithm ('validation data'). The second bucket contains over 165, 000 requests and is used to evaluate various algorithms ('test data'). Each request is accompanied with 130 playlists. The features are constructed from both user and playlist information. The user features include demographic information such as gender, age and geographic information such as region, language, time; as well as affinity scores computing the degree to which the user likes the content based on historic interactions. For playlists, features include platform, product, genre, type and tag information. Constant augmentation and feature normalisation as in [7] are applied.

Policy Evaluation: We follow the unbiased offline evaluator in [20], and compare the proposed approach with baselines. Appendix C details the policy evaluation setup in detail.

5.3 Reward based evaluation

We begin the comparison of different methods by investigating how the approaches compare on the GGF of rewards obtained using the *simulated dataset*. Reward based comparison allows us to compare model performance based on how high the obtained reward is for the different approaches. Since we are trying to maximise GGF, higher GGF values imply better algorithm. We plot the mean GGF of the average reward from various algorithms along with their standard deviations in Figures 2 wherein we vary the number of objectives from $D = 5$ to $D = 20$ and vary the number of arms from $K = 50$ to $K = 500$ respectively. The different subplots present results wherein D and K are varied.

First, we observe that when the mean reward $\boldsymbol{x}_{[t]}$ changes at each round t due to different contexts, the recently proposed multi-objective bandit approach (MO-OGDE baseline; non-contextual) performs no better than random selection; which highlights the importance of leveraging context; thereby motivating the need for multi-objective variants of contextual bandits. Second, the mixed strategy always performs better than the pure strategy, which highlights the benefit of selecting an arm based on a probability distribution, rather than selecting only the single best arm each time in a deterministic fashion. This result highlights the benefit of exploration-exploitation trade-off; i.e., enabling the model to trade-off exploitation of known arms with exploration of potentially useful arms, is beneficial.

Third, we see that all algorithms that involve some element of uncertainty, e.g. ϵ-greedy, LinUCB and MO-LinCB, perform better than the mixed strategy approach, which solves a surrogate of the true objective exactly. Therefore, stochastic algorithms are advantageous. Finally, our proposed algorithm, MO-LinCB, performs better than all compared approaches, including ϵ-greedy and LinUCB over the entire range of D and K values investigated, which demonstrates the utility of optimizing the Gini-function formulation of different objectives in a contextual bandit setting.

5.4 Optimizing for Multiple User Metrics

We use the *Music Streaming Dataset* to investigate performance on the task of optimizing multiple user centric objectives. In Figure 3, we plot the reward distributions of the objectives clicks, total length streamed and total songs played for the different approaches.

We observe that all strategies that leverage contextual information significantly improve upon random selection, which asserts that the constructed features possess predictive power for all objectives and the online ridge regression is effective. From the results of single-objective algorithms, we observe that ϵ-greedy for CTR not only performs best for CTR but also for the other two user centric metrics (L & S). This can be explained by the fact that the constructed features have more predictive power for clicks and that there is correlation between the objectives.

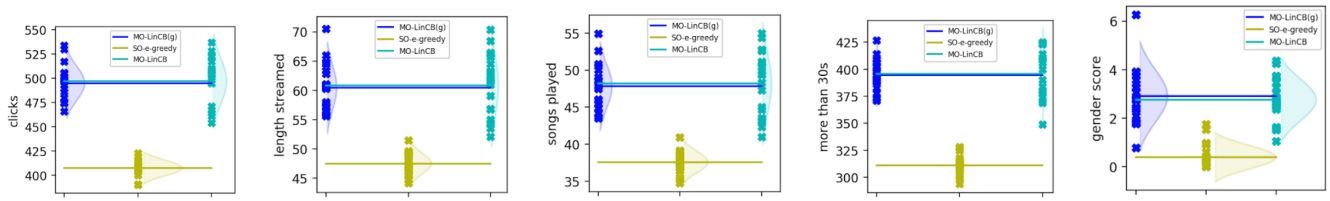

Figure 4: [Music data test results] Comparison of single objective and multi-objective model for the different objectives upon adding a gender based promotion objective.

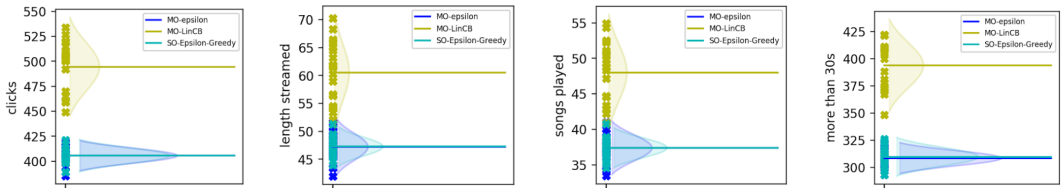

Figure 5: [Music data test results] Investigating the difference in performance across different approaches of multi-objective optimization. The results highlight that the way of doing Multi-objective matters.

Upon comparing the single-objective and multi-objective algorithms, we observe that both the multi-objective ϵ-greedy ϵ-g (MO) and proposed MO-LinCB algorithms perform better than all the single-objective algorithms (ϵ-g (C),ϵ-g (S), ϵ-g (L)) with much less variance. This agrees with a known result that fusing information from correlated objectives is beneficial [1, 16], thereby implying that single-objective algorithms are not Pareto-efficient, and that jointly optimising for correlated objectives is better than individually optimising for them separately.

Most importantly, we observe that the proposed MO-LinCB approach outperforms all other approaches, with over 6.9%, 11.0% and 10.0% gains across the three different metrics, respectively. This highlights the fact that the proposed MO-LinCB is better than ϵ-g (MO) by using guided rather than random exploration.

These results show that optimizing for multiple user interaction metrics performs better for each metric than directly optimizing that metric alone. Considering multiple user interaction metrics has enabled the model to have a holistic view of user experience and the model benefits from the inherent chaining structure exhibited by the metrics considered (i.e. click → stream → #songs streamed).

5.5 Impact of Promotional Objective

Beyond users, recommender systems often have other stakeholders, e.g. artists or retailers. While the experiments so far considered multiple user centric objectives, many competing objectives exist, e.g. local vs global artists, popularity, gender. In this paper, we use gender to illustrate our approach on promotional objectives. The correlation analysis (Figure 1) shows a correlation of -0.12 between promotion and satisfaction objective, which highlights that the promotion objective is a mildly competing objective to satisfaction.

We work with the real world music streaming data (as described in Section 5.2.2), but for each recommended set, we additionally consider the gender of all the artists whose tracks belong to this set, and compute the proportion of non-male artist, which we refer to as gender metric. Figure 4 presents the results for the four user satisfaction metrics as well as the gender metric comparing performance of the single objective ϵ-greedy approach with variants of the proposed multi-objective linear bandit models: with and without the gender objective. Similar to the previous result, we observe that both variants of the proposed multi-objective models perform better than the single objective case, across all five metrics. Furthermore, we observe gains in gender metric when we add gender as an additional optimization objective. It is important to note that the gain in gender objective was achieved *without* hurting any of the four satisfaction metrics. This hints at the fact that the proposed MO-LinCB is able to find better pareto-optimal solutions than other approaches, thereby increasing performance across all metrics.

This result highlights that optimizing for promotional metrics in addition to user satisfaction metrics need not be a zero-sum game; i.e., it may be possible to achieve gains in promotional metrics without hurting satisfaction metrics. One possible explanation for this might be the fact that a large proportion of users indeed want to listen to more gender-diverse music, and introducing gender-diversity as an additional metric satisfies those users more.

5.6 Comparing Multi-optimization Methods

The results presented above highlight the importance of considering many objectives while optimizing recommendation systems. We now investigate the importance of the specific approach adopted to optimize multiple metrics using the real world music streaming data. We question whether any approach for multi-objective model would work. In Figure 5 we compare the proposed method with multi-objective extension of the widely used $\epsilon-$greedy bandit approach, as well as single objective bandit model. We observe that the multi-objective variant of $\epsilon-$greedy bandit approach performs comparably with the single objective bandit model, whereas the proposed multi-objective bandit model performs better than both baselines. This highlights the fact that the specific approach of leveraging multiple objectives matters; and that naive multi-objective extensions may not perform as well.

6 CONCLUSIONS

Recommender systems powering multi-stakeholder platforms often need to consider different stakeholders and their objectives when serving recommendations. Based on correlation analysis across the

different stakeholder objectives, we highlighted that trade-offs exist between objectives, and motivated the need for multi-objective optimization of recommender systems. To address this problem, we presented MO-LinCB, a multi-objective linear contextual bandit model, which leverages Gini aggregation function to scalarize multiple objectives, and proposed a scalable gradient ascent based approach to learn the recommendation policy. The GGI based method helps the model to not only minimize the cumulative cost for each objective, but also balance the different objectives.

Our findings suggest that optimizing for multiple objectives is often beneficial to all objectives involved. We observed that optimizing for multiple satisfaction metrics result in improved performance for each satisfaction metric, which highlights the fact that different satisfaction metrics capture different aspects of user behavior, and jointly considering them results in surfacing better recommendations. Furthermore, it is possible to obtain gains in both complementary and competing objectives via multi-objective optimization. Detailed experiments around quantifying the impact of competing objectives on satisfaction metrics highlight the benefits offered by the proposed model. The proposed approach was able to obtain gains in a competing promotional objective, without hurting user satisfaction metrics.

While our findings have implications on the design of recommender system powering multi-stakeholder platforms, there exist a number of future research directions to pursue. First, one may be interested in having more control over the importance of different objectives. While Gini function favors equitable distribution, other functions could be researched so that to provide fine-grained control over objectives. Second, a possible extension could include global objectives that are not session-specific but aggregated over time. Third, the ability to optimize for multiple metrics motivates the need to quantify other objectives of stakeholders. Finally, we only experimented with a subset of possible objectives. It will be important to extend this subset with a wide range of objectives from various stakeholders, in particular those very competing objectives.

REFERENCES

[1] Gediminas Adomavicius and YoungOk Kwon. 2007. New Recommendation Techniques for Multicriteria Rating Systems. *IEEE Intelligent Systems* (2007).

[2] Deepak Agarwal, Bee-Chung, Pradheep, and Xuanhui. 2011. Click shaping to optimize multiple objectives. In *Proceedings of KDD 2011*.

[3] Deepak Agarwal, Bee-Chung Chen, Pradheep Elango, and Xuanhui Wang. 2012. Personalized click shaping through lagrangian duality for online recommendation. In *Proceedings of SIGIR 2012*.

[4] Leon Barrett and Srini Narayanan. 2008. Learning All Optimal Policies with Multiple Criteria. In *ICML*. 41–47.

[5] Róbert Busa-Fekete, Balázs Szörényi, Paul Weng, and Shie Mannor. 2017. Multi-objective Bandits: Optimizing the Generalized Gini Index. In *ICML*.

[6] Konstantina Christakopoulou, Jaya Kawale, and Arindam Banerjee. [n.d.]. Recommendation with capacity constraints. In *Proceedings of CIKM 2017*.

[7] Wei Chu and Seung-Taek Park. 2009. Personalized Recommendation on Dynamic Content Using Predictive Bilinear Models. In *WWW*.

[8] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of Recommender Algorithms on Top-n Recommendation Tasks. In *RecSys*.

[9] M. M. Drugan and A. Nowe. 2013. Designing multi-objective multi-armed bandits algorithms: A study. In *International Joint Conference on Neural Networks (IJCNN)*.

[10] Zoltán Gábor, Zsolt Kalmár, and Csaba Szepesvári. 1998. Multi-criteria Reinforcement Learning. In *ICML*. 197–205.

[11] Rupesh Gupta, Guanfeng Liang, Ravi Kiran Tseng, Xiaoyu Chen, and Romer Rosales. [n.d.]. Email volume optimization at LinkedIn. In *KDD 2016*.

[12] Tamas Jambor and Jun Wang. [n.d.]. Optimizing multiple objectives in collaborative filtering. In *Proceedings of RecSys 2010*.

[13] Dietmar Jannach and Malte Ludewig. 2017. When recurrent neural networks meet the neighborhood for session-based recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*. ACM, 306–310.

[14] Thorsten Joachims. 2002. Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 133–142.

[15] I. Y. Kim and O. L. de Weck. 2006. Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation. *Structural and Multidisciplinary Optimization* 31, 2 (2006), 105–116.

[16] Anisio Lacerda. 2015. Contextual Bandits for Multi-objective Recommender Systems. In *Proceedings of the 2015 Brazilian Conference on Intelligent Systems (BRACIS)*. 68–73.

[17] Kleanthi Lakiotaki, Nikolaos F Matsatsinis, and Alexis Tsoukias. 2011. Multicriteria user modeling in recommender systems. *IEEE Intelligent Systems* (2011).

[18] John Langford and Tong Zhang. 2008. The Epoch-Greedy Algorithm for Multi-armed Bandits with Side Information. In *NIPS*.

[19] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. 2010. A Contextual-bandit Approach to Personalized News Article Recommendation. In *WWW*.

[20] Lihong Li, Wei Chu, John Langford, and Xuanhui Wang. 2011. Unbiased Offline Evaluation of Contextual-bandit-based News Article Recommendation Algorithms. In *WSDM*.

[21] C. Liu, X. Xu, and D. Hu. 2015. Multiobjective Reinforcement Learning: A Comprehensive Overview. *IEEE Transactions on Systems, Man, and Cybernetics* (2015).

[22] Donald W. Marquardt and Ronald D. Snee. 1975. Ridge Regression in Practice. *The American Statistician* 29, 1 (1975), 3–20.

[23] James McInerney, Benjamin Lacker, Samantha Hansen, Karl Higley, Hugues Bouchard, Alois Gruson, and Rishabh Mehrotra. [n.d.]. Explore, Exploit, and Explain: Personalizing Explainable Recommendations with Bandits. In *Proceedings of RecSys 2018*.

[24] Rishabh Mehrotra and Benjamin Carterette. 2019. Recommendations in a marketplace. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 580–581.

[25] Rishabh Mehrotra, James McInerney, Hugues Bouchard, Mounia Lalmas, and Fernando Diaz. 2018. Towards a Fair Marketplace: Counterfactual Evaluation of the trade-off between Relevance, Fairness & Satisfaction in Recommendation Systems. In *CIKM*.

[26] K. Van Moffaert, K. Van Vaerenbergh, P. Vrancx, and A. Nowe. [n.d.].

[27] Eric Moulines and Francis R. Bach. 2011. Non-Asymptotic Analysis of Stochastic Approximation Algorithms for Machine Learning. In *NIPS*.

[28] Włodzimierz Ogryczak and Tomasz Sliwinski. 2003. On solving linear programs with the ordered weighted averaging objective. *European Journal of Operational Research* 148, 1 (2003), 80 – 91.

[29] Saba Q. Yahyaa, Madalina M. Drugan, and Bernard Manderick. 2014. Knowledge Gradient for Multi-objective Multi-armed Bandit Algorithms. In *Proceedings of the 6th International Conference on Agents and Artificial Intelligence - Volume 1*.

[30] Marco Tulio Ribeiro, Lacerda, Veloso, and Ziviani. [n.d.]. Pareto-efficient hybridization for multi-objective recommender systems. In *RecSys 2012*.

[31] Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. *The Annals of Mathematical Statistics* 22, 3 (1951), 400–407.

[32] Lili Shan, Lei Lin, and Chengjie Sun. [n.d.]. Combined Regression and Tripletwise Learning for Conversion Rate Prediction in Real-Time Bidding Advertising. In *SIGIR 2018*.

[33] Aleksandrs Slivkins. 2014. Contextual Bandits with Similarity Information. *J. Mach. Learn. Res.* 15, 1 (2014), 2533–2568.

[34] Richard S. Sutton and Francis Bach. 1998. *Reinforcement Learning: An Introduction*. MIT Press.

[35] Krysta M Svore, Maksims N Volkovs, and Christopher JC Burges. [n.d.]. Learning to rank with multiple objective functions. In *Proceedings of WWW 2011*.

[36] Cem Tekin and Eralp Turğay. 2018. Multi-objective contextual multi-armed bandit with a dominant objective. *IEEE Transactions on Signal Processing* (2018).

[37] Eralp Turğay, Doruk Öner, and Cem Tekin. [n.d.]. Multi-objective contextual bandit problem with similarity information. *arXiv preprint arXiv:1803.04015, 2018* ([n. d.]).

[38] Umair ul Hassan and Edward Curry. 2016. Efficient task assignment for spatial crowdsourcing: A combinatorial fractional optimization approach with semi-bandit learning. *Expert Systems with Applications* 58 (2016), 36–56.

[39] John A. Weymark. 1981. Generalized Gini inequality indices. *Mathematical Social Sciences* 1 (1981), 409–430.

[40] Lin Xiao, Minand, Zhaoquan, L Yiqun, and Ma Shaoping. [n.d.]. Fairness-aware group recommendation with pareto-efficiency. In *RecSys 2018*.

[41] R. R. Yager. [n.d.]. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, & Cybernetics* ([n. d.]).

[42] Saba Yahyaa, Madalina Drugan, and Bernard Manderick. 2015. Thompson Sampling in the Adaptive Linear Scalarized Multi Objective Multi Armed Bandit. In *Proceedings of the 7th International Conference on Agents and Artificial Intelligence*.

[43] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. [n.d.]. Beyond clicks: dwell time for personalization. In *Proceedings of RecSys 2014*.

[44] Chongjie Zhang and Julie A Shah. 2014. Fairness in Multi-Agent Sequential Decision-Making. In *NIPS*. 2636–2644.

Algorithm 2 Policy Evaluator

Input: stream of events $\mathbb{S}$, bandit algorithm Ξ, desired number of events τ
Set $\mathbb{H}_0 = \emptyset$ (an initially empty history)
Set $\tilde{\boldsymbol{x}}_{[0]} = \mathbf{0}$ (an initially zero total reward)
Set $c = 0$ (an initially zero counter for registered events)
for $t = 1, 2, 3, \ldots$ **do**
  if $c = \tau$ **then**
    Stop
  else
    Get next event $(\mathcal{J}_{[t]}, k, \boldsymbol{x}_{[t]})$ from $\mathbb{S}$
    if $\Xi(\mathcal{J}_{[t]}, \mathbb{H}_c) = k$ **then**
      $\mathbb{H}_{c+1} = \mathbb{H}_c \cup \{(\mathcal{J}_{[t]}, k, \boldsymbol{x}_{[t]})\}$
      $\tilde{\boldsymbol{x}}_{[c+1]} = \tilde{\boldsymbol{x}}_{[c]} + \boldsymbol{x}_{[t]}$
      $c = c + 1$
    else
      Skip
Output: $f(\tilde{\boldsymbol{x}}_{[\tau]})$, e.g. $G_{\boldsymbol{w}}(\frac{1}{\tau}\tilde{\boldsymbol{x}}_{[\tau]})$

A APPENDIX A: DETAILS OF SIMULATED DATASET

Here we describe in detail the exact setting and parameters used to create the simulated dataset. To generate random multi-objective contextual bandit problems, we draw the universal parameter, ϑ^*, uniformly from $[0, 1)^{M \times D}$. Each element of the feature with length M, $\mathcal{J}_{[t],k}$, for each arm k at each round t, is chosen independently from $\mathcal{N}(\frac{1}{M}, \frac{1}{M^2})$. Each element of the noise $\boldsymbol{\zeta}_{[t],d}$ for each objective d at each round t is generated according to $\mathcal{N}(0, 0.01\boldsymbol{\mu}_d^2)$, where $\boldsymbol{\mu}_d = \boldsymbol{f}_{[t]}^T \vartheta_d^*$ is the mean reward for objective d of the arm chosen at round t. We set M to 10 and run a total of $T = 10,000$ rounds. We set the regularisation parameter, λ to $0.1 \in [0, 1]$ [22], which is sufficient to ensure that the estimated $\hat{\vartheta}$ converges to the actual ϑ^* as more rounds are played. The number of arms K is set to $\{50, 100, 200, 500\}$ and the number of objectives is taken from $D \in \{5, 10, 20\}$. We set the weight vector $\boldsymbol{w}$ of GGF to $\boldsymbol{w}_d = 2^{-d+1}, 1 \leq d \leq D$, the same as in [5].

B APPENDIX B: IMPLEMENTATION DETAILS

We experimented with different values of exploration probability ϵ for the ϵ-greedy baseline, $\epsilon \in [1\%, 5\%, 10\%]$ and picked the best performing estimate (ϵ=1%) for comparison. Similarly, for LinUCB and MO-OGDE, δ is set to 10% following empirical comparison for different tolerance values, $\delta \in [1\%, 5\%, 10\%]$. The step size η is set to 5 and the number of iterations I is set to 5 for MO-LinCB, following empirical evaluation on various step sizes and iterations. We also study single-objective ϵ-greedy algorithms using each of the three objectives denoted by ϵ-g (C), ϵ-g (L) and ϵ-g (S) for the click, length and stream objectives respectively. Each algorithm is repeated 30 times. $\epsilon = 0.01$ is used in all ϵ-greedy algorithms in music data experiments as well.

C APPENDIX C: POLICY EVALUATION

Given a bandit algorithm and a desired number of user interaction sessions τ, we step through the stream of recorded sessions sequentially. If it happens that the algorithm selects the same arm as the one that is recorded via an i.i.d. uniformly random policy, we register such an event and update our algorithm and the total payoff. If the algorithm selects a different arm, we ignore such an event and move on to the next event without changing the state of the algorithm. The detailed procedure is presented in Algorithm 2. While this evaluation approach suffers from inefficient data usage, we defer a more detailed counterfactual evaluation for future work. We use this dataset in Sections 5.4 for evaluating the impact on multiple user satisfaction metrics, Section 5.5 for quantifying the impact on promotional objective and in Section 5.6 for comparing the different multi-objective models.

	MO-LinCB			LinUCB			MO-OGDE		
time/s	K=50	K=200	K=500	K=50	K=200	K=500	K=50	K=200	K=500
D=5	0.46	0.85	1.68	0.52	1.43	3.75	0.26	0.67	1.5
D=10	0.47	0.88	1.72	0.63	1.86	4.67	0.26	0.69	1.52
D=20	0.52	0.98	1.87	0.84	2.68	6.55	0.28	0.71	1.6

Table 1: Running times of different approaches.

clicks					
Algorithms	Random	e-g (C)	e-g (L)	e-g (S)	e-g (MO)
MO-LinCB	<0.1%	<0.1%	<0.1%	<0.1%	<0.1%
e-g (MO)	<0.1%	<0.1%	<0.1%	<0.1%	
e-g (S)	<0.1%	<0.1%	11.8%		
e-g (L)	<0.1%	<0.1%			
e-g (C)	<0.1%				

Table 2: P-values of 'clicks' for all pairs of algorithms.

D APPENDIX D: SCALABILITY

We measured the running time of different algorithms as K and D vary using the simulation data experiments and observed that pure and mixed strategies were orders of magnitude slower, which render them impractical. Running times for remaining approaches are summarise in Table 1. Our proposed MO-LinCB is roughly three times faster than LinUCB at large K. This is crucial in recommendation as more choices (higher K) enables better personalisation.

E APPENDIX E: SIGNIFICANCE TEST

To test for real differences between the mean metrics of different approaches, we assume that scores from any algorithm are normally distributed,[3] and present the p-values under the null hypothesis (there is no difference between the mean scores between any two algorithms) in Table 2 for the click metric, where unbiased estimators for the population variances are used. We see that the differences between the results of relevant algorithms are statistically significant.

[3]This is a good approximation by central limit theorem given the adequate data size.