A Multi-objective Optimization Framework for Multi-stakeholder Fairness-aware Recommendation

HAOLUN WU*, School of Computer Science, McGill University, Canada
CHEN MA*, Department of Computer Science, City University of Hong Kong, Hong Kong SAR
BHASKAR MITRA, Microsoft Research, Canada
FERNANDO DIAZ, Google, Canadian CIFAR AI Chair, Canada
XUE LIU, School of Computer Science, McGill University, Canada

Nowadays, most online services are hosted on multi-stakeholder marketplaces, where consumers and producers may have different objectives. Conventional recommendation systems, however, mainly focus on maximizing consumers' satisfaction by recommending the most relevant items to each individual. This may result in unfair exposure of items, thus jeopardizing producer benefits. Additionally, they do not care whether consumers from diverse demographic groups are equally satisfied. To address these limitations, we propose a multi-objective optimization framework for fairness-aware recommendation, *Multi-FR*, that adaptively balances accuracy and fairness for various stakeholders with Pareto optimality guarantee. We first propose four fairness constraints on consumers and producers. In order to train the whole framework in an end-to-end way, we utilize the smooth rank and stochastic ranking policy to make these fairness criteria differentiable and friendly to back-propagation. Then, we adopt the multiple gradient descent algorithm to generate a Pareto set of solutions, from which the most appropriate one is selected by the Least Misery Strategy. The experimental results demonstrate that *Multi-FR* largely improves recommendation fairness on multiple stakeholders over the state-of-the-art approaches while maintaining almost the same recommendation accuracy. The training efficiency study confirms our model's ability to simultaneously optimize different fairness constraints for many stakeholders efficiently.

1 INTRODUCTION AND MOTIVATION

When viewed from a sociotechnical lens, conventional deployed machine learning systems demonstrate a range of socially problematic behaviors, including algorithmic bias and misinformation [18]. Multisided recommendation systems in marketplaces, content-distribution networks, and matchmaking platforms require reasoning about the potential impact on several populations of stakeholders with potentially disparate and possibly conflicting objectives. As such, the potential societal implications of a recommendation algorithm must balance multiple objectives across these groups.

Two of the most important stakeholders in these multisided systems are (i) producers who provide goods and services, and (ii) consumers who purchase them. When a recommendation system systematically overlooks the utility of certain historically disadvantaged groups, inequity can be exacerbated for producers and consumers. We take the movie recommendation platform as an example, where young children are likely to be a minority of consumers and they favor cartoons. If the system is solely concerned with the utility of adults in order to maximize revenues, this may have a detrimental effect on the satisfaction of young children, and hence on the utility of cartoon producers. In this circumstance, the cartoons receive insufficient exposure, and their creators may leave the platform as a result of poor earnings. Such an unbalanced market will work against those

*Corresponding Author.

Authors' addresses: Haolun Wu, haolun.wu@mail.mcgill.ca, School of Computer Science, McGill University, 845 Sherbrooke St W, Montreal, Quebec, Canada, H3A 0G4; Chen Ma, chenma@cityu.edu.hk, Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Y6302, Yeung Kin Man Academic Building, Kowloon Tong, Hong Kong SAR; Bhaskar Mitra, bhaskar.mitra@microsoft.com, Microsoft Research, 2000 McGill College Ave, Montreal, Quebec, Canada, H3A 3H3; Fernando Diaz, diazf@acm.org, Google, Canadian CIFAR AI Chair, 1253 McGill College Ave, Montreal, Quebec, Canada, H3B 2Y5; Xue Liu, xueliu@cs.mcgill.ca, School of Computer Science, McGill University, 845 Sherbrooke St W, Montreal, Quebec, Canada, H3A 0G4.

Table 1. Properties of different recommendation approaches. Our proposed method, Multi-FR, can model both the consumer-sided fairness and producer-sided fairness jointly in one recommendation framework while incorporating sensitive attributes. Our fairness metrics are also differentiable and friendly for back-propagation. Most significantly, our strategy is theoretically guaranteed to be Pareto optimal with respect to all objectives and does not need handcraft tuning on the scaling factors.

Approaches & Collections	Consumer-sided Fairness	Producer-sided Fairness	Using Sensitive Attributes	Adaptive Factor Learning	Differentiable Metrics	Pareto Optimal
Accuracy-centric Policy [40, 42, 46, 47, 56, 76]	✗	✗	**N.A.**	**N.A.**	✗	**N.A.**
Consumer-sided Fairness [24, 30, 43, 70]	✔	✗	✔/✗	✗	✗	✗
Producer-sided Fairness [1, 6, 53, 59, 72, 73, 85]	✗	✔	✔/✗	✗	✗	✗
Two-sided Fairness [58, 75, 81]	✔	✔	✔/✗	✗	✗	✗
Multi-FR (Ours)	✔	✔	✔	✔	✔	✔

adults' utility if they ever wish to watch cartoons someday, thus degrading the overall experience on the platform.

Unfortunately, existing approaches for recommendation systems/platforms are stuck in the aforementioned issue. First, conventional accuracy-centric approaches employ various data-driven models [40, 42, 46, 47, 56, 76] to estimate the relevance scores of the consumer-product pair, and then recommend the top-K most relevant products to the corresponding consumers. However, these approaches can create a huge disparity in the exposure of the producers due to the "superstar economics" [3, 53], which is unfair for the producers and also harms the health of the marketplace. Second, many approaches merely address either the consumer-sided fairness [24, 30, 43, 70] or the producer-sided fairness [1, 6, 59, 72, 73, 85], but neglect the fairness on the other side. This is not desirable, as producers and consumers are both indispensable in these marketplaces. Third, several approaches [58, 81] consider the fairness on both sides merely at an individual level (e.g., envy-freeness), but ignore the sensitive attributes (e.g., age, gender, popularity) at a group level. We regard this as sub-optimal, since the society strives to not overlook the utility of minority groups in the real world. Thus, the sensitive attributes should be taken into consideration.

Apart from the limitation that no prior works have attempted to model both the consumer-sided and producer-sided fairness at a group level concurrently in a recommendation framework, another disadvantage is that prior approaches [53, 75] on fairness-aware recommendation generally require fine-tuning weights for multiple objectives in order to optimize the overall objective, which is tedious and cannot guarantee a satisfactory final solution. When involving multiple objectives, one optimal solution is that no objective can be further improved without impairing the others. This optimality is widely recognized and is referred to as the *Pareto optimality* [50]. Existing approaches for optimizing multiple objectives can be broadly classified into two categories: (i) heuristic search and (ii) scalarization (weighted summation). While multi-objective evolutionary algorithms are frequently used in heuristic search, they ensure that the emerging solutions are not dominated by each other (but can still be dominated by Pareto optimal solutions) [39]. Thus they cannot guarantee Pareto optimality. The scalarization method converts multiple objectives into a single objective with weighted summation and can achieve Pareto optimal solutions with proper scalarization [31]. However, existing approaches that manually adjust the scaling factors for each objective frequently fail to satisfy the necessary conditions for Pareto optimality. As a result, it is still a challenge for current methods in achieving Pareto optimality when optimizing multiple fairness objectives.

To address the aforementioned problems, we treat the multi-stakeholder fairness-aware recommendation as a multi-objective optimization (MOO) task and propose a scalable framework, namely *Multi-FR*. We summarize the differences between prior works and our work in Table 1. As shown, our work is the first one that satisfies all the properties. Specifically, we propose a method to differentiate four fairness metrics on the consumer side and the producer side, taking into account different attributes within this framework. Thus, the fairness constraints can be directly optimized through back-propagation. We adopt the weighted summation to combine all these fairness constraints as well as the objective for the recommendation accuracy into one unified framework. Thereafter, the multiple gradient descent algorithm (MGDA) along with the Frank-Wolfe Solver [29, 69] are utilized to generate scaling factors regarding these fairness and recommendation objectives for satisfying the necessary conditions of Pareto optimality. It is worthy to notice that the scaling factors can be adaptively updated during the training procedure without handcraft tuning and the whole framework is trained in an end-to-end way. Finally, the most appropriate solution is selected by the Least Misery Strategy [44, 60]. The experimental results on three public datasets indicate that our method can well balance the recommendation quality and fairness, and significantly outperform other state-of-the-art fairness-aware recommendation methods on all the fairness metrics.

To summarize, our contributions are:

- We propose a general fairness-aware recommendation framework with the multi-objective optimization, *Multi-FR*, which jointly optimizes accuracy and fairness for consumers and producers in an end-to-end way. The final solution is guaranteed to be Pareto optimal theoretically.
- We leverage the multiple gradient descent algorithm with the Frank-Wolfe Solver to guarantee that the scaling factors satisfy the necessary conditions of Pareto optimality. Furthermore, these scaling factors are updated adaptively throughout the training procedure, eliminating the need of manual-crafted search.
- We propose a method for differentiating the fairness criteria on both the consumer and producer sides through utilizing the smooth rank and stochastic ranking policy, so that these fairness constraints can be optimized directly and are friendly to back-propagation.
- Extensive experimental results on three public datasets comparing with three state-of-the-art fairness-aware recommendation approaches demonstrate that *Multi-FR* largely improves the recommendation fairness with little drop in terms of the recommendation quality. Further analysis indicates the capability of *Multi-FR* in terms of optimizing multiple fairness objectives concurrently with efficiency.

2 RELATED WORK

In this section, we provide a summary regarding the related studies from the following three aspects: (i) the definition of fairness in recommendation systems, (ii) approaches to achieving fairness, and (iii) recommendation with multiple objectives. We close this section by highlighting the novelty and difference of our work compared to prior methods.

2.1 Definition of Fairness in Recommendation

Prior works on fairness in recommendation, from the perspective of stakeholders, consider algorithmic effects on consumers (i.e. users who seek content) and producers (i.e. users who provide content), independently or together.

On the consumer side, fairness refers to systematic differential performance [52] across consumers. Some works focus on the individual fairness that ensures similar individuals are treated similarly [23]. Other works define fairness on a group level and aim to make the system provide

comparable quality of service or utility to consumers within different demographic groups (e.g., gender, race, age) [25]. Chaney et al. [17] demonstrate, through simulation, that feedback loops inherent in the production system can exacerbate unfairness and homogenize recommendation. Yao and Huang [84] demonstrate that these issues can be addressed by introducing fairness constraints during the training process.

Other works focus on fairness on the producer side, whose fairness can be defined as the systematic differential exposure [6, 21, 53, 73] across content producers and, most often, groups of producers (e.g. grouped by genre or popularity). For instance, Ekstrand et al. [26] find that standard recommendation algorithms may result in certain demographic groups of producers being over- or under-represented in recommendation decisions. Beutel et al. [5] demonstrate that these issues can be addressed in production systems by defining pairwise fairness objectives and introducing them as learning objectives.

Joint satisfaction for consumer-sided fairness and producer-sided fairness is an important requirement for a healthy marketplace. To capture fairness for multisides, Burke et al. [13] introduce the task of two-sided fairness and employ the sparse linear method (SLIM) [57] to address it. Patro et al. [58] borrow notions from fair division [74] to model the two-sided fairness in recommendation. Specifically, they ensure the envy-freeness-up-to-one on the consumer side and maximin share guarantee of exposure on the producer side [12]. Both fairness is treated at an individual level. Wu et al. [81] follow a similar way to model the individual fairness on both sides, where they ensure each individual consumer obtains equal satisfaction and each individual producer obtains equal (or proportional) exposure. Both of the above two works adopt algorithms similar to Round-robin scheduling [2, 8, 15] to achieve the fairness in recommendation. Sühr et al. [75] experiment with two-sided fairness in the context of ride-hailing platforms, where the two-sided objective is a linear interpolation of consumer and producer fairness metrics, like other works.

2.2 Approaches to Achieving Fairness

Motivated by the idea of constructing multiple objectives in recommendation [36, 51], most works on fairness in recommendation and ranking scenarios model the fairness as an extra loss. It works as a supplement to the accuracy (quality) loss in the whole objective function [72, 73, 82], followed by employing the scalarization technique. It is expected to achieve a Pareto optimal recommendation [65, 66] when multiple objectives are concerned. However, existing studies mostly depend on manually assigning weights for scalarization, whose Pareto optimality cannot be guaranteed.

Recent studies have proposed to use the adversarial learning and causal graph reasoning techniques to achieve fairness in recommendation systems. For instance, Beigi et al. [4] propose an adversarial learning-based recommendation model with attribute protection, which can protect consumers from the private-attribute inference attack while simultaneously recommending relevant products to consumers. Rahman et al. [63] find that the bias in recommendation is caused by unfair graph embeddings and thus propose a novel fairness-aware graph embedding algorithm *Fairwalk* to achieve the statistical parity. Bose and Hamilton [9] combine the adversarial training with the graph representation learning together to protect sensitive features of consumers. They introduce an adversarial framework to enforce fairness on graph embeddings. Similarly, Wu et al. [79] propose a graph based adversarial learning method, *FairGO*, to filter any sensitive information hidden in the data representation, where the fairness requirement is defined as not to expose sensitive features during the user modelling. The benefits of these algorithms lie in explicitly modelling the fairness into the representation embeddings; however, the models are based on more advanced techniques and they do not consider multisided fairness explicitly.

Fair Learning-to-Rank (LTR) is another popular research direction for achieving fairness in the community nowadays, and several recent works have raised the question of group fairness in

rankings. Zehlike et al. [85] formulate the problem as a "Fair top-K ranking" that aims to guarantee the occurrences of items within the protected group is above a minimum threshold in every prefix of the top-K ranking list based on some pre-defined proportion. Celis et al. [16] propose a constrained maximum weight matching algorithm for ranking a set of items efficiently under a fairness constraint indicating the maximum number of items with each sensitive attribute being allowed in the top positions. Most recently, some works break the parity constraints restricting the fraction of items with each attribute in the ranking but extend the LTR methods to a large class of possible fairness constraints. For instance, Biega et al. [6] aim to achieve amortized fairness of attention by making exposure proportional to relevance through integer linear programming. Singh et al. [72] propose a more general framework that can achieve both individual fairness and group fairness solutions via a standard linear program and the Birkhoff-von Neumann decomposition [7].

2.3 Recommendation with Multiple Objectives

The studies on multi-objective optimization are rich and various approaches have been proposed [19]. One significant characteristic of multi-objective optimization is that, usually, there does not exist a solution that satisfies all the objectives simultaneously.

Some studies have considered multiple objectives in personalized recommendation tasks [36, 65]. For instance, Ribeiro et al. [65] construct multiple objectives including accuracy, diversity, and novelty. And then a Pareto frontier is found to satisfy the mentioned objectives. However, manual scalarization (grid search) is still required. Besides, to the best of our knowledge, there are few studies on optimizing multiple objectives in group recommendation and we are among the first to address the multi-stakeholder fairness problem in recommendation from the multi-objective optimization perspective.

Novelty and Difference to Prior Works. Our study expands on prior works by studying interpolation-free optimization of multi-stakeholder fairness problems. Compared to prior works, we highlight the novelty and difference of our work as follows. Firstly, we are the first to propose a general framework for multi-stakeholder fairness-aware recommendation with theoretical guarantee that the final solution is Pareto stationary (Pareto optimal under mild assumptions). Prior works hardly satisfy the Pareto optimality. Secondly, we propose a way to differentiate the fairness metrics on both the consumer side and the producer side, so that we can directly optimize these fairness constraints during the model training in an end-to-end way. However, most metrics defined in prior works are not differentiable. Thus, they can only use a post-processing method to audit the recommendation list after obtaining the relevance scores between users and items. Thirdly, we employ the Frank-Wolfe Solver and propose a method to learn the scaling factors on multiple objectives adaptively during training. However, most prior works require additional scaling factor tuning. In addition, we utilize the stochastic ranking policy, which is capable of distributing the exposure among producers more equitably, whereas most prior works adopt the static ranking.

3 PRELIMINARIES

3.1 List of Notations

The notations we used in this paper are shown in Table 2.

3.2 Problem Formulation

We consider a top-K item recommendation task in this paper which takes the user implicit feedback as input. We denote the set of all users and items as $\mathcal{U}$ and $\mathcal{I}$, respectively. For each user u, the user preference data is represented by a set of items he/she has interacted with as $\mathcal{I}_u^+ := \{i \in \mathcal{I} | Y_{u,i} = 1\}$ where $\boldsymbol{Y} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$ is the binary implicit feedback rating matrix. We then split $\mathcal{I}_u^+$ into a training set

Table 2. Description of Notations.

Notations	Description
$\mathcal{U}, \mathcal{I}$	The set of users and items
$\mathcal{D}_u$	The set of training triplets
$\mathcal{L}_i$	The i^{th} objective
Θ	Embeddings for users and items
α	The scaling factor on each objective
b, K	Batch size and the maximum length of the recommendation list
n^C, n^P	Number of groups on the consumer side and the producer side
t	The total number of objectives in the model
m, n	The number of fairness constraints on the consumer side and the producer side
l_i	The relevance score between the i^{th} item and the current query (user)
r_i	The rank of the i^{th} item w.r.t. the current query
$\tilde{r}_i$	The smooth rank of the i^{th} item w.r.t. the current query under the stochastic policy
γ	User's patience factor that controls the depth of browsing a list of items
τ	Temperature that controls the smoothness of ranks under the stochastic policy
$\boldsymbol{Y} \in \mathbb{R}^{\lvert\mathcal{U}\rvert \times \lvert\mathcal{I}\rvert}$	The binary relevance matrix between items and users
$\boldsymbol{s}^{G_i} \in \mathbb{R}^K$	The satisfaction representation for group G_i on the consumer side
$\boldsymbol{m}^{G_i} \in \mathbb{R}^b$	The mask of group G_i on the consumer side
$\boldsymbol{N} \in \mathbb{R}^{b \times K}$	The matrix of NDCG@1 to NDCG@K for all the users in a batch
$\epsilon, \epsilon^* \in \mathbb{R}^{n_p}$	System exposure and target exposure on the producer side
$\boldsymbol{R} \in \mathbb{R}^{b \times \lvert\mathcal{I}\rvert}$	Ranks for all items with respect to a batch of users
$\boldsymbol{E}, \boldsymbol{E}^* \in \mathbb{R}^{b \times \lvert\mathcal{I}\rvert}$	System and target exposure matrix of all items w.r.t. a batch of users
$\boldsymbol{c} \in \mathbb{R}^{\lvert\mathcal{I}\rvert}$	Group label of all items

$\mathcal{S}_u^+$ and a test set $\mathcal{T}_u^+$, requiring that $\mathcal{S}_u^+ \cup \mathcal{T}_u^+ = \mathcal{I}_u^+$ and $\mathcal{S}_u^+ \cap \mathcal{T}_u^+ = \emptyset$. Then the top-$K$ recommendation is formulated as: given the training item set $\mathcal{S}_u^+$, and the non-empty test item set $\mathcal{T}_u^+$ of user u, the model aims to recommend an ordered set of K items $\mathcal{X}_u$ such that $|\mathcal{X}_u| = K$ and $\mathcal{X}_u \cap \mathcal{S}_u^+ = \emptyset$.

As aforementioned, our goal of this work is to well balance all of the following objectives on the quality and fairness in an end-to-end recommendation framework. We employ the notion of *group fairness* [23] on both the consumer side and the producer side, which suggests treating different groups in a fair way.

(1) *Recommendation Accuracy*: This is to ensure that the recommendation model is capable of capturing consumers' real preference. The recommendation quality can be evaluated by a matching score between $\mathcal{T}_u^+$ and $\mathcal{X}_u$, such as Recall@K or NDCG@K (Normalized Discounted Cumulative Gain).
(2) *Consumer-sided Fairness*: This is to guarantee that the consumers belonging to different demographic groups receive the same level of satisfaction. Specifically, we aim to minimize the group-level satisfaction difference between any two groups of consumers. The definition will be detailed in Section 4.
(3) *Producer-sided Fairness*: This is to avoid the producers belonging to minority groups receive an extremely high/low opportunity of being exposed. Specifically, we aim to minimize the difference between the current computed exposure (system exposure) and the ideal exposure (target exposure) of producers. The definition will be detailed in Section 4.

4 OBJECTIVE CONSTRUCTION

In this section, we demonstrate the formulation of each objective: recommendation quality, consumer-sided fairness, and producer-sided fairness, followed by offering the technical details for making all of these objectives differentiable for an end-to-end training [1].

4.1 Objective for Recommendation Quality

In order to ensure the recommendation quality, we adopt the Bayesian Pairwise Ranking (BPR) model proposed by Rendle et al. [64]. Denoting those items that are unobserved by user u in $\mathcal{S}_u^+$ as $\mathcal{S}_u^- := \mathcal{I} \setminus \mathcal{S}_u^+$, we define a new training set in which each component is a triplet:

$$\mathcal{D}_u := \left\{(u, i^+, i^-) | i^+ \in \mathcal{S}_u^+ \wedge i^- \in \mathcal{S}_u^-\right\} . \tag{1}$$

Then the goal of the recommendation model is to generate a total ranking $>_u$ of all items for each user u. The binary relation $>_u$ is required to be a total order on the set of items $\mathcal{I}$. The relation $i^+ >_u i^-$ specifies that user u prefers item i^+ over item i^-. Thereby, we aim to maximize:

$$p\left(\Theta | \{>_u\}_{\mathcal{D}_u}\right) \propto p\left(\{>_u\}_{\mathcal{D}_u} | \Theta\right) p(\Theta) , \tag{2}$$

where $\Theta = [\![\Theta^U, \Theta^I]\!]$ is the model parameter containing the user and item embeddings, and $\{>_u\}_{\mathcal{D}_u}$ denotes the observed preferences in the training data. We aim to identify the parameters Θ that maximize this posterior over all users and all pairs of items. Assuming that users act independently, we have:

$$p\left(\{>_u\}_{\mathcal{D}_u} | \Theta\right) = \prod_{(u,i^+,i^-) \in \mathcal{D}_u} p\left(i^+ >_u i^- | \Theta\right) . \tag{3}$$

We define the probability that a user prefers item i^+ over item i^- as:

$$p\left(i^+ >_u i^- | \Theta\right) = \sigma\left(\hat{x}_{ui^+}(\Theta) - \hat{x}_{ui^-}(\Theta)\right) , \tag{4}$$

where $\hat{x}_{ui}(\Theta) = \langle \Theta_u^U, \Theta_i^I \rangle$, denoting the inner product between two embeddings. If we adopt a normal distribution as the prior for $p(\Theta)$, then we can formulate the optimization objective as:

$$\begin{aligned} \mathcal{L}^{Accuracy} &= \operatorname*{argmax}_{\Theta} \ln p\left(\Theta | \{>_u\}_{\mathcal{D}_u}\right) \\ &= \operatorname*{argmin}_{\Theta} \sum_{(u,i^+,i^-) \in \mathcal{D}_u} -\ln \sigma\left(\hat{x}_{ui^+}(\Theta) - \hat{x}_{ui^-}(\Theta)\right) + \lambda_\Theta ||\Theta||_2^2. \end{aligned} \tag{5}$$

We can optimize this via stochastic gradient descent by repeatedly drawing triples (u, i^+, i^-) randomly from the training set and updating the model parameter Θ.

4.2 Fairness Objectives for Multi-stakeholder

Fairness attracts more attention in current information retrieval systems, which has a huge impact on the multi-stakeholder marketplaces. In our proposed framework, we resort to group fairness on both the consumer (user) side and the producer (item) side. We define four fairness constraints (two for each side) with respect to different attributes in our model which are summarized in Table 3.

[1] In this paper, we use *producer-sided fairness* to represent the fairness on the item side, although the producer information is not provided in most datasets. There are two reasons for choosing this term: (1) We would like to make the term consistent with prior works [53, 75, 81] in community; (2) People are building systems for humans not for items: achieving fairness on items actually benefits the producers. We will collect the supplier/producer information of items in future work. Additionally, *user* and *consumer* share the same meaning throughout the remainder of this paper.

Table 3. Four attribute-based fairness constraints on the consumer side and the producer side.

Stakeholder	Attribute-based Fairness Constraints	
Consumer	**Gender**-based	**Age**-based
Producer	**Popularity**-based	**Genre**-based

4.2.1 Consumer Fairness Constraint

It has been shown that sensitive features affect the satisfaction of consumers in recommendation [87]. For instance, Ekstrand et al. [25] and Neophytou et al. [55] demonstrate that recommendation performance can vary across demographic groups and Mehrotra et al. [52] report similar observations in the context of web search. For addressing this, the fairness of consumers is generally defined as the fairness in quality of service, such as ensuring those consumers belonging to different demographic groups (with different sensitive features) experience comparable recommendation quality. In this work, we follow the similar line to model the group fairness on the consumer side. Specifically, we aim to make the satisfaction of different groups with sensitive features be ideally equal. Here we adopt the NDCG@K, a widely-used ranking metric, to measure the satisfaction of consumers.

However, since items ranked at higher positions generally receive more attention from consumers, we argue that the fairness at each prefix of the ranking is also significant. As a result, solely considering the recommendation equality for an entire ranking list is not enough. Thereby, we construct the $\mathbf{s}^{G_i} \in \mathbb{R}^K$ as a satisfaction vector for the group G_i among the consumers, where the k^{th} entry of $\mathbf{s}^{G_i}$ equals to NDCG@k, $k = 1, ..., K$. Assuming there are n^C groups among the consumer side, the fairness constraint on a group level can be defined as the mean of the pair-wise satisfaction difference across all groups:

$$\mathcal{L}^{C-Fair} = \frac{1}{\binom{n^C}{2}} \sum_{1 \le i < j \le n^C} \left\| \mathbf{s}^{G_i} - \mathbf{s}^{G_j} \right\|_2^2 , \tag{6}$$

where $\binom{n^C}{2}$ is the number of combinations between different groups.

More specifically, considering a batch of b users (consumers), we define the satisfaction of group G_i as:

$$\mathbf{s}^{G_i} = \frac{\mathbf{m}^{G_i} \cdot \mathbf{N}}{||\mathbf{m}^{G_i}||_1} , \tag{7}$$

where $\mathbf{m}^{G_i} \in \mathbb{R}^b$ is a multi-hot vector, $\mathbf{N} \in \mathbb{R}^{b \times K}$ is a matrix containing NDCG@1 to NDCG@K for all users in a batch. The method of computing the NDCG values in a differentiable way is detailed in section 4.3. The denominator is used to compute the number of users in the group G_i.

It is worthy to notice that any kinds of attributes can be adopted for defining the mask $\mathbf{m}^{G_i}$ in Eq. 7. In this work, we focus on two types of disparities regarding the two most common and sensitive attributes on the consumer side.

Gender-based Fairness. Gender is one of the most sensitive attributes of humans and many works have already presented insightful observations and analysis on gender bias in Internet services [14, 37]. Thus, we construct the gender mask $\mathbf{m}^{G_1}$ and $\mathbf{m}^{G_2}$ for females and males and aim to minimize the satisfaction difference between these two groups. *Note that gender is treated as a binary class due to the available labels in the datasets. We do not intend to suggest that gender identities are binary, nor support any such assertions.*

Age-based Fairness. Other than gender, we also consider fairness with respect to the age attribute. We split the age into 7 stages (0-17, 18-24, 25-34, 35-44, 45-49, 50-55, 56+) following the

criterion in the MovieLens datasets [33] and construct $\mathbf{m}^{G_{a_i}}$ as the mask for the i-th age group. Then optimizing the age-based fairness is to minimize the difference of satisfaction for all age groups, as described in Eq. 6.

4.2.2 Producer Fairness Constraint

Previous works in recommendation mainly assume that the users are the only stakeholder in a recommendation system; however, the items should also be taken into consideration since they represent the benefits of the producers [58, 75], which are an equally significant stakeholder in a commerce marketplace. Thus, we model the producer-sided fairness to guarantee the satisfaction of producers.

In comparison to consumers, producers are more concerned with profits in these marketplaces, which are highly related to the exposure of their products/items. Unfair exposure distribution on items may make certain producers unsatisfied and hence leave the platform. This may reduce the market's diversity, which in turn may harm the consumers' utility. Therefore, for the group fairness on producers, the goal is to find a ranking strategy that can offer a fair probability of exposure on items based on their merits. However, one of the key challenges, as mentioned in [21], is that a single fixed ranking for a query (in retrieval) or user (in recommendation) tends to limit the ability of an algorithm to distribute exposure amongst relevant items. For a static ranking, (i) some relevant items may receive more exposure than other relevant items, and (ii) some irrelevant items may receive more exposure than other irrelevant items. Therefore, we aim to find a policy that samples a permutation from a distribution over the set of all permutations of $|\mathcal{I}|$ items, and such a stochastic ranking policy should be able to force all items to receive a fair exposure proportional to their merits, thus achieving fairness with respect to the exposure of items in expectation.

Assuming there are n^P groups among all items, the fairness constraint on the producer side can be defined as the difference between two exposure vectors:

$$\mathcal{L}^{P-Fair} = \|\epsilon - \epsilon^*\|_2^2, \tag{8}$$

where $\epsilon \in \mathbb{R}^{n^P}$ is a vector representing the distribution of exposure on items from different groups, $\epsilon^* \in \mathbb{R}^{n^P}$ is the ideal exposure distribution proportional to the true relevance of items. We refer to the ϵ and ϵ^* as the *system exposure* and *target exposure* of the system, respectively. Hereafter, we demonstrate how to model ϵ and ϵ^*.

For computing the system exposure ϵ, we need to first rank all the $|\mathcal{I}|$ items for b users in a batch based on the relevance score. Under a static ranking policy, this is generally achieved by sorting the items in a descending order based on the predicted preference score between users and items. We can thus obtain a matrix $\boldsymbol{R} \in \mathbb{R}^{b\times|\mathcal{I}|}$ containing the ranks of all items for b queries (users). (As noted before, we are interested in a stochastic policy instead. The method for obtaining the matrix $\boldsymbol{R}$ under a stochastic policy will be detailed in section 4.3.) To compute the exposure, we adopt a well-known user browsing model, the position-based model [21, 54], that assumes a user's probability of visiting a position decreases exponentially with the rank. Then we compute the exposure of all items in a batch as $\boldsymbol{E} = \gamma^{\boldsymbol{R}} \in \mathbb{R}^{b\times|\mathcal{I}|}$, where γ represents the patience factor that controls how deep a user is likely to browse in a ranked item list.

We denote the group label on all items in $\mathcal{I}$ sorted by the item id as $\boldsymbol{c} \in \mathbb{R}^{|\mathcal{I}|}$, where each entry contains the group label that the corresponding item belongs to. However, the items are displayed in different orders for each user, we thus denote the group label of items ranked for the i^{th} user

based on the ranking order as $\boldsymbol{c}^i \in \mathbb{R}^{|\mathcal{I}|}$, which is a permutation of $\boldsymbol{c}$. Then, the ϵ is computed as:

$$\epsilon_k = \frac{\sum_{i=1}^{b} \sum_{\boldsymbol{c}_j^i = k} \boldsymbol{E}_{ij}}{b \cdot ||\mathbb{1}_{\boldsymbol{c}_j = k}||_1}. \tag{9}$$

Here the $\mathbb{1}_{\boldsymbol{c}_j=k}$ is the indicator vector, where a "1" at position j refers to $\boldsymbol{c}_j = k$ and a "0" otherwise. The k^{th} entry of ϵ contains the average exposure on all items belonging to the k^{th} group.

As for the target exposure ϵ^*, we assume all relevant items should have the same high probability of being selected to the top of the ranking, while other irrelevant items should equally share the rest amount of exposure at a low level. Given the binary relevance label in the training set as $\boldsymbol{Y} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$, we assume the number of relevant items for each query (user) is t_i, which is obtained through counting the number of ones in each row of $\boldsymbol{Y}$. Then the target exposure of all items in a batch can be computed as:

$$\boldsymbol{E}^*_{ij} = \begin{cases} \frac{1}{t_i} \sum_{m \in [1, t_i]} \gamma^m, & \text{if } \boldsymbol{Y}_{ij} = 1, \\ \frac{1}{|\mathcal{I}| - t_i} \sum_{m \in [t_i+1, |\mathcal{I}|]} \gamma^m, & \text{otherwise.} \end{cases} \tag{10}$$

Thus, we can construct the target exposure in a similar way as Eq. 9:

$$\epsilon^*_k = \frac{\sum_{i=1}^{b} \sum_{\boldsymbol{c}_j^i = k} \boldsymbol{E}^*_{ij}}{b \cdot ||\mathbb{1}_{\boldsymbol{c}_j = k}||_1}. \tag{11}$$

Then we can use Eq. 8, Eq. 9, and Eq. 11 to optimize for the producer-sided fairness.

We consider two types of group fairness constraints on the producer side. Both are based on the same framework defined in Eq. 8, while the only difference lies in the different construction of the group label $\boldsymbol{c}$ during the modelling.

Popularity-based Fairness. The "superstar economics" [3, 53] always occurs in real-world recommendation scenarios, where a small number of most popular artists/items/products possess most of the attention of consumers. A major side-effect of the "superstar economics" is the impedance to producers on the tail-end of the spectrum, who struggle to attract consumers and are not satisfied with the marketplace.

To construct the popularity-based group label, we rank all the $|\mathcal{I}|$ items based on their occurrences in the dataset from the highest to the lowest and evenly split them into 5 groups labelled from 4 to 0, where each group contains 20% of items.

Genre-based Fairness. We do not expect any specific genre of items to receive too much or too little exposure in a marketplace; therefore, genre-based fairness is also worthy of taking into consideration. We only use the MovieLens datasets to investigate this fairness and make use of all the 18 movie genres in the datasets [33].

4.3 Differentiable Approximation of the Ranking

In our formulation, relevance is defined as a function of item rankings, but the sorting operation is inherently non-differentiable. To mitigate this problem, we adopt the continuous approximation of the ranking function proposed in [62, 80] that is amenable to gradient-based optimization. The key

insight behind these approximations lies in defining the rank of an item in terms of the pairwise preference with every other item in the collection:

$$r_i = 0.5 + \sum_{j}^{n} \sigma'(l_j - l_i), \text{ where } \sigma'(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0.5, & \text{if } x = 0 \\ 0, & \text{if } x < 0 \end{cases}. \tag{12}$$

Here the l_i is the relevance score between the i^{th} item to the query (user), which is computed through the inner-product between the user embedding and the item embedding. The discrete function $\sigma'(\cdot)$ is typically approximated using the differentiable sigmoid function.

Given the approximated differentiable ranks of items, it is then straightforward to derive an optimization objective for standard relevance metrics—*e.g.*, discounted cumulative gain (DCG)—that can be directly optimized using the gradient descent. Assuming that we consider the items in a ranking up to position K, then the SmoothDCG is defined as:

$$\text{SmoothDCG} = \sum_{i=1}^{K} \frac{l_i}{\log_2(r_i + 1)}. \tag{13}$$

Therefore, we adopt the normalized version of such SmoothDCG when constructing the fairness objective in Eq. 7 on the consumer side during training, but still, use the original NDCG in the evaluation phase.

As for the producer side, in order to mitigate the similar non-differentiable operation when constructing the ranking position matrix $\boldsymbol{R}$ and also adopt a stochastic ranking policy rather than a static one, we use the Plackett-Luce model [61] for constructing a ranking by sampling items sequentially, followed by adopting the Gumbel Softmax technique proposed in [11, 48]. For a single query (user), we recall that the sampling probability of an item by using a static Plackett-Luce policy [61] is:

$$p_i = \frac{\exp(l_i)}{\sum_{j \in \mathcal{I}} \exp(l_j)}. \tag{14}$$

As aforementioned, the static policy will limit the ability of the ranking algorithm for fairly distributing the exposure. Thus, what the stochastic ranking policy specifically performs here is: (i) reparameterizing the probability distribution by adding independently-drawn noise ζ sampled from the Gumbel distribution to l and (ii) sorting items by the "noisy" probability distribution $\tilde{p}_i$:

$$\tilde{p}_i = \frac{\exp(l_i + \zeta_i)}{\sum_{j \in \mathcal{I}} \exp(l_j + \zeta_j)}. \tag{15}$$

However, the sorting operation itself is non-differentiable either. To address this, we instead compute the smooth rank [80] for each item in the ranking as follows:

$$\tilde{r}_i = \sum_{j \in \mathcal{I}, j \neq i} \left(1 + \exp\left(\frac{\tilde{p}_i - \tilde{p}_j}{\tau}\right)\right)^{-1}, \tag{16}$$

where the temperature τ is a hyperparameter that controls the smoothness of the approximated ranks. Then the exposure $\boldsymbol{E}$ in Eq. 9 is computed based on this smooth rank.

We now achieve a differentiable way for modelling both the consumer-sided fairness and the producer-sided fairness for an end-to-end training.

5 MULTI-FR: MULTI-STAKEHOLDER FAIRNESS-AWARE RECOMMENDATION

In this section, we introduce our proposed framework, *Multi-FR*, for fairness-aware recommendation in multi-stakeholder marketplaces. In conventional recommendation systems, the main aim lies in satisfying consumers. However, it has been shown in recent studies [58] that solely optimizing the satisfaction of consumers may jeopardize the benefits of producers who are essential participants in two-sided markets. Thus, to achieve a personalized, satisfactory, and fair recommendation simultaneously in one joint framework is significant in both academia and industry.

Traditionally, these aspects are modelled as specific objectives and combined by summation with manually set scaling factors. However, utilizing hand-crafted factors has two major drawbacks. First, these scaling factors incur tedious hyper-parameter tuning. This would cost many trials and substantial computation resources to identify appropriate scaling factors, especially when the number of objectives is huge. Second, each objective in the summed objective function may need a different magnitude of scaling values in the training process. Setting one fixed value is not capable of dynamically balancing all of these objectives well during the training process.

To tackle the aforementioned problems, we treat the fairness-aware recommendation as a multi-objective optimization problem and propose a framework for optimizing multiple objectives jointly. Before diving into the final framework, we start from providing some major techniques and theoretical guarantees in the multi-objective optimization in order to better illustrate the entire picture.

5.1 Multi-Objective Optimization

A multi-objective optimization task is usually defined as optimizing a set of possibly conflicting objectives. Given a set of objectives, the MOO aims to find a solution that can optimize all objectives simultaneously:

$$\min_{\theta} \mathcal{L}(\theta) = \min_{\substack{\theta^c \\ \theta^{s_1},...,\theta^{s_t}}} \mathcal{L}(\theta^c, \theta^{s_1}, ..., \theta^{s_t}) = \min_{\substack{\theta^c \\ \theta^{s_1},...,\theta^{s_t}}} \begin{bmatrix} \mathcal{L}_1(\theta^c, \theta^{s_1}) \\ \mathcal{L}_2(\theta^c, \theta^{s_2}) \\ \vdots \\ \mathcal{L}_t(\theta^c, \theta^{s_t}) \end{bmatrix}^\intercal, \tag{17}$$

where $\mathcal{L}$ is the full objective, $\mathcal{L}_1$, ..., $\mathcal{L}_t$ are t different objectives, respectively. θ^c is the common parameter shared by all objectives, while θ^{s_1}, ..., θ^{s_t} are the objective-specific parameters.

Notice that one of the key characteristics of an MOO problem is that a solution that can optimize each objective to an ideal situation may not exist. This is exactly due to the conflict and correlation among the objectives as discussed before. The optimal solution of an MOO problem should balance all the objectives, which is called the Pareto optimality.

Definition 1. Pareto Optimality

(1) A solution θ_1 **dominates** another solution θ_2 if for all objectives $\mathcal{L}_i(\theta_1^c, \theta_1^{s_i}) \leq \mathcal{L}_i(\theta_2^c, \theta_2^{s_i})$, where $i \in \{1, ..., t\}$, and there exists at least one objective $j \in \{1, ..., t\}$, where $\mathcal{L}_j(\theta_1^c, \theta_1^{s_j}) < \mathcal{L}_j(\theta_2^c, \theta_2^{s_j})$.
(2) A solution is of **Pareto optimality** if there does not exist any other solution that dominates it. In this case, we also call the solution is **Pareto optimal**.
(3) There is usually more than one solution being Pareto optimal in an MOO problem. The set of such solutions is called **Pareto set**, which is the solution set of an MOO problem. The curve of the points in the Pareto set is called the **Pareto frontier**.

5.2 Multiple Gradient Descent Algorithm

The multiple gradient descent algorithm (MGDA) [20] is one of the most effective methods for MOO. It can reach an optimal point for all objectives with theoretical guarantee. Borrowing the idea from the gradient descent on a single objective, the MGDA can be regarded as an extension of the gradient-based algorithm on multiple objectives. The overall objective of solving an MOO problem by MGDA is usually a weighted summation of t single objectives, defined as:

$$\mathcal{L}(\theta) = \mathcal{L}(\theta^c, \theta^{s_1}, ..., \theta^{s_t}) = \sum_{i=1}^{t} \alpha_i \cdot \mathcal{L}_i(\theta^c, \theta^{s_i}), \tag{18}$$

where the coefficients of all the objectives satisfy $\sum_{i=1}^{t} \alpha_i = 1$ and $\alpha_i \geq 0$, for $i = 1, ..., t$.

It is hard to find direct conditions for Pareto optimality. Therefore, we introduce the Pareto stationarity, which is a necessary condition for Pareto optimality in an MOO problem [20, 28, 67]. A Pareto optimal solution must be Pareto stationary, while the reverse may not hold.

Definition 2. Pareto Stationarity
A solution θ^* is of **Pareto stationarity** if it satisfies all of the following conditions:

(1) $\sum_{i=1}^{t} \alpha_i = 1, \alpha_i \geq 0$, for $i = 1, ..., t$,
(2) $\sum_{i=1}^{t} \alpha_i \nabla_{\theta^{*c}} \mathcal{L}_i(\theta^{*c}, \theta^{*s_i}) = 0$,
(3) $\nabla_{\theta^{*s_i}} \mathcal{L}_i(\theta^{*c}, \theta^{*s_i}) = 0$, for $i = 1, ..., t$.

The above conditions are also known as the **Karush-Kuhn-Tucker (KKT) conditions**, which was first proposed by Kuhn and Tucker [41].

The MGDA leverages the KKT conditions to solve the MOO problem, which are necessary for optimality. Sener and Koltun [69] propose to solve a quadratic-form constrained minimization problem defined as follows:

$$\begin{aligned} \min_{\alpha_1, \alpha_2, ..., \alpha_t} & \left\| \sum_{i=1}^{t} \alpha_i \cdot \nabla_{\theta^c} \mathcal{L}_i(\theta^c, \theta^{s_i}) \right\|_2^2, \\ \text{s.t.,} & \sum_{i=1}^{t} \alpha_i = 1, \alpha_i \geq 0, \text{for } i = 1, ..., t. \end{aligned} \tag{19}$$

Given Eq. 19, there are two situations for the final solution: the final solution is Pareto stationary if the solution to this optimization problem makes the Euclidean norm equals to 0; otherwise, the solution offers a common descent direction which benefits all the objectives as proved by [20]. Therefore, one can use the single-objective gradient descent for optimizing the objective-specific parameters θ^{s_i} on t different objectives and employ the obtained solution to the above equations for updating the common parameters θ^c.

5.3 Solving the MOO Problem

We first introduce a special case where there are only two objectives in the loss function:

$$\min_{\alpha \in [0,1]} \|\alpha \cdot \nabla_{\theta^c} \mathcal{L}_1(\theta^c, \theta^{s_1}) + (1 - \alpha) \cdot \nabla_{\theta^c} \mathcal{L}_2(\theta^c, \theta^{s_2})\|_2^2 . \tag{20}$$

The analytical solution to this quadratic problem is:

Algorithm 1: Frank-Wolfe Solver [29, 35, 69]

Input: $t \leftarrow$ number of objectives
$\theta \leftarrow$ model parameters: $(\theta^c, \theta^{s_1}, ..., \theta^{s_t})$
Output: A list of learned scaling coefficients: $\alpha_1, ..., \alpha_t$

1 Random Initialization: $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_t)$ satisfying the constraints in Eq. 19.
2 Precompute $\boldsymbol{M}$, $\forall i, j \in \{1, ..., t\}$:
3 $\quad \boldsymbol{M}_{ij} = (\nabla_{\theta^c} \mathcal{L}_i(\theta^c, \theta^{s_i}))^\top (\nabla_{\theta^c} \mathcal{L}_j(\theta^c, \theta^{s_j}))$
4 **repeat**
5 $\quad i^* = \text{argmin}_r \sum_i \alpha_i \boldsymbol{M}_{ri}$
6 $\quad w^* = \text{argmin}_w (w\boldsymbol{e}_{i^*} + (1-w)\boldsymbol{\alpha})^\top \boldsymbol{M} (w\boldsymbol{e}_{i^*} + (1-w)\boldsymbol{\alpha})$ ←Using Procedure 1
7 $\quad \boldsymbol{\alpha} = w^* \boldsymbol{e}_{i^*} + (1-w^*)\boldsymbol{\alpha}$ $\quad (\boldsymbol{e}_{i^*}$ is the unit vector)
8 **until** w^* *converge or maximum iteration reaches*;
9 **return** $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_t)$

Procedure 1: Solving $\text{argmin}_{w \in [0,1]} ||wx_1 + (1-w)x_2||_2^2$

1 $\quad w^* = \frac{(x_2 - x_1)^\top x_2}{||x_1 - x_2||_2^2}$
2 $\quad w^* = \max(\min(w^*, 1), 0)$
3 $\quad$ **return** w^*;

$$\alpha^* = \frac{(\nabla_{\theta^c} \mathcal{L}_2(\theta^c, \theta^{s_2}) - \nabla_{\theta^c} \mathcal{L}_1(\theta^c, \theta^{s_1}))^\top \nabla_{\theta^c} \mathcal{L}_2(\theta^c, \theta^{s_2})}{\|\nabla_{\theta^c} \mathcal{L}_1(\theta^c, \theta^{s_1}) - \nabla_{\theta^c} \mathcal{L}_2(\theta^c, \theta^{s_2})\|_2^2}, \tag{21}$$

where the α^* should be clipped into $[0, 1]$ as $\alpha^* = \max(\min(\alpha, 1), 0)$.

Although there are no analytical solutions for more than two objectives in an MOO problem, we can still borrow the analytical solution of two objectives to conduct the line search efficiently. This technique was proposed by Jaggi [35] to accelerate the convergence of the Frank-Wolfe algorithm [29]. Specifically, as shown in Algorithm 1, the procedure with the same idea of Eq. 21 is regarded as a subroutine to compute the w^* on line 6. The scaling factors generated by the Frank-Wolfe Solver satisfy the KKT conditions aforementioned.

5.4 The Multi-FR Framework and the Overall Objective

Thereby, we propose the *Multi-FR* framework to utilize the scaling factors that satisfy the KKT conditions to generate Pareto stationary (can be regarded as Pareto optimal under mild assumptions in real world) solutions which can smoothly balance the recommendation quality and multisided fairness. The algorithm is shown in Algorithm 2. It is worthy to mention that all the three tasks in this work described in Section 3.2 require the same parameters (user embeddings and item embeddings) for computing the relevance scores between users and items, and there are no task-specific parameters. Thus, we omit θ^s and set the model parameter $\theta = \theta^c = \Theta = [\![\Theta^U, \Theta^I]\!]$. As a result, we generate all objectives using the unified parameter, which is the Θ including user and item embeddings.

For building the overall training objective, we use the weighted summation framework following [27, 38, 83] since this is the most common choice in optimizing multiple objectives. The weighted summation is a convex combination of objectives and each single objective optimization determines one particular optimal solution point on the Pareto frontier. For ensuring the final solution is Pareto stationary (Pareto optimal under mild assumptions), we adopt Algorithm 1 to adaptively compute the scaling factors $\boldsymbol{\alpha}$ that satisfy the convex constraints in KKT conditions as aforementioned. Thus, our overall training objective can be formulated as follows:

Algorithm 2: Multi-FR Framework

Initialization( )
for $i \in 1, ..., t$ **do**
  Construct the individual objective $\mathcal{L}_i(\Theta)$ for each task
  (The objective for recommendation quality and the fairness constraints)
end
for $epoch \in 1, ..., n_{epoch}$ **do**
  for $batch \in 1, ..., n_{batch}$ **do**
    Forward_Passing( )
    for $i \in 1, ..., t$ **do**
      Compute the gradient for each objective: $\nabla_\Theta \mathcal{L}_i(\Theta)$
      Gradient_Normalization( ) (optional)
    end
    $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_t)$ ← Using Algorithm 1
    Construct the single aggregated objective: $\mathcal{L}(\Theta) = \sum_{i=1}^{t} \alpha_i \cdot \mathcal{L}_i(\Theta)$
    $\nabla_\Theta \mathcal{L}(\Theta) = \sum_{i=1}^{t} \alpha_i \cdot \nabla_\Theta \mathcal{L}_i(\Theta)$
    Update Θ
  end
end
return Θ that can lead to Pareto stationarity / Pareto optimality on quality and fairness

$$
\begin{aligned}
\mathcal{L} =& \alpha^A \cdot \mathcal{L}^{Accuracy} + \sum_{i=1}^{m} \alpha_i^C \cdot \mathcal{L}_i^{C-Fair} + \sum_{j=1}^{n} \alpha_j^P \cdot \mathcal{L}_j^{P-Fair}, \\
\text{s.t., } & \alpha^A + \sum_{i=1}^{m} \alpha_i^C + \sum_{j=1}^{n} \alpha_j^P = 1. \\
& \alpha^A \geq 0, \alpha_i^C \geq 0, \alpha_j^P \geq 0, \forall i, j.
\end{aligned}
\tag{22}
$$

Here, m and n refer to the number of fairness constraints on the consumer and producer sides, respectively.

It is worth noticing that our proposed *Multi-FR* framework does not rely on specific formulations of the loss functions or the model structures. Although the afore four disparity measures belong to the group fairness, one can also define any individual fairness objectives and scalably integrate them into our framework as long as they are differentiable.

5.5 Solution Selection

There is no consensus strategy on choosing one single Pareto optimal solution from a Pareto set since any solution in the Pareto set cannot strictly dominate the others. To select a proper solution, we borrow the idea from one of the most well-known metrics in Theoretical Economics, the *Least Misery Strategy* [60], for guiding us to select a "fair" solution for all the objectives. Lin et al. [44] also adopt this criteria to select the final Pareto optimal solution. The main idea is that we do not want the worst objective to be too bad.

Table 4. The statistics of datasets.

	MovieLens100K	MovieLens1M	FM-reduced
#User	943	5,805	19,234
#female/#male/#age	273/670/7	1,655/4,150/7	4,022/15,212/7
#Item	1,682	3,574	9,703
#genre/#popularity	18/5	18/5	-/5
#Interaction	100,458	678,740	1,049,322
Density	6.33%	3.27%	0.56%

Motivated by the *Least Misery Strategy*, our final solution aims to choose the solution coming from the q'^{th} round that minimizes the highest loss value across all the objectives:

$$\min_{1 \le q' \le q} \max\{\mathcal{L}_1^{q'}, \mathcal{L}_2^{q'}, ..., \mathcal{L}_t^{q'}\}, \tag{23}$$

where t is the total number of objectives in the model and q is the total number of running rounds. The t here can be defined as any positive integer depending on the number of objectives to model in regarding to the consumer-sided fairness and the producer-sided fairness. The superscript q' indicates the q'^{th} round. Therefore, given a generated Pareto frontier by running Algorithm 1 and Algorithm 2 for q rounds ($q = 5$) in our proposed model, we pick the solution coming from the round q' with the minimum value of Eq. 23 as the final recommendation.

6 EXPERIMENTS

In this section, we evaluate the proposed model and other baseline methods on three real-world datasets.

6.1 Datasets

The proposed model is evaluated on three real-world datasets from various domains with different sparsities: *MovieLens100K* [2], *MovieLens1M* [3] [33], and *FM-reduced* [4]. *MovieLens100K* and *MovieLens1M* are user-movie datasets collected from the *MovieLens* website. These two datasets provide 100 thousand and 1 million user-movie interactions, respectively, with the user metadata (gender and age group) and movie genres. The *FM-reduced* dataset is collected from the last.fm website, which contains the music listening records of 360 thousand users along with the gender and age of users. The original version of this dataset is too large to run the majority of previously developed fairness-aware algorithms, as it would take a huge consumption both in time and space. In order to make the experiments able to be conducted on all baselines, we reduce the size of the dataset by first randomly selecting 25,000 users. Under the implicit feedback setting, we keep those ratings no less than four (out of five) as positive feedback and treat all other ratings as missing entries for all datasets. To filter noisy data, we only keep the users with at least ten ratings and the items at least with five ratings.

We adopt the age group strategy of the MovieLens dataset to split users into 7 different age groups and the movies into 18 different genres in all experiments. For all the datasets, we also group the items into 5 different groups based on their popularity. For each user, we randomly split 70%, 10%, 20% of the rated items as the training set, validation set, and testing set, respectively. The statistics for the datasets after preprocessing are shown in Table 4.

[2]https://grouplens.org/datasets/movielens/100k/

[3]https://grouplens.org/datasets/movielens/1m/

[4]http://ocelma.net/MusicRecommendationDataset/lastfm-360K.html

6.2 Evaluation Metrics

In this section, we demonstrate our chosen metrics on the recommendation accuracy, fairness, and diversity. We adopt both self-defined metrics and commonly used measurements in academia. Our measurement of fairness and diversity covers the *individual level*, *group level*, and *system level*.

Metrics for measuring recommendation accuracy:

- **Recall@**K, which indicates the percentage of her rated items that appear in the top-K recommended items. The greater the value of this metric, the higher the quality of the model.
- **NDCG@**K, which is the normalized discounted cumulative gain at K, which takes the position of correctly recommended items into account. The greater the value of this metric, the higher the quality of the model.

Self-defined Metrics for measuring fairness:

- **Disparity**$_u$ measures the unfairness on the user side, i.e. Eq. 6. The smaller the value of this metric, the higher the consumer-sided fairness of the model.
- **Disparity**$_i$ measures the unfairness on the item side, i.e. Eq. 8. The smaller the value of this metric, the higher the producer-sided fairness of the model.

General metrics for measuring fairness and diversity:

- **Gini Index** measures the inequality among values of a frequency distribution [22], e.g., numbers of occurrences (exposures) in the recommendation list. This measurement is at *individual level*. Given a list of exposure of all items ($\mathcal{I}$) aggregated over all the recommendation lists, $l_e = [e_1, e_2, ..., e_{|\mathcal{I}|}]$, the Gini Index is calculated as below [5]:

$$\text{Gini}(l_e) = \frac{1}{2|\mathcal{I}|^2\overline{e}} \sum_{i=1}^{|\mathcal{I}|} \sum_{j=1}^{|\mathcal{I}|} |e_i - e_j|, \tag{24}$$

  where $\overline{e}$ is the mean of all item exposures. The smaller the value of the Gini Index, the higher the fairness of the model.
- **Popularity rate** computes the proportion of popular items in the recommendation list against the total number of items in the list, which can be regarded as a *group-level* measurement. The smaller the value of the popularity rate, the higher the fairness of the model.
- **Simpson's diversity index** was introduced in 1949 by Edward H. Simpson to measure the degree of concentration when individuals are classified into types [71]. This is suitable for the scenario of recommendation since items are generally categorized into different groups. This metric was also adopted before by Zhou et al. [86] for measuring the diversity in recommendation. Therefore, we employ this metric as a *system-level* measurement of diversity that takes into account the number of groups present, as well as the relative abundance of each group. Given a list of exposures of all items in the recommendation results and the group label of each, the Simpson's diversity index can be formulated as:

$$\text{Diversity} = 1 - \left(\frac{\sum_{i=1}^{g} n_i(n_i - 1)}{N(N-1)}\right), \tag{25}$$

  where g is the total number of groups, n_i is the total number of items of group i, and N is the total number of items of all groups. This diversity index can also be interpreted as the probability that two randomly sampled items (without replacement) do not belong to the same group. The greater the value of this metric, the higher the diversity of the system.

[5]It is worthy to notice that, for Gini Index, there exist multiple alternative expressions [49, 77]. Here we adopt the most widely used version as demonstrated in book [68] and paper [30]. The main difference between these definitions lies in whether to compute the coefficient with direct reference to the Lorenz curve [45].

6.3 Method Studied

We choose three models as our recommendation backbones:

- **BPRMF**, Bayesian Personalized Ranking-based Matrix Factorization [64], which is a classic method for learning pairwise personalized rankings from user implicit feedback.
- **WRMF**, Weighted Regularized Matrix Factorization [34], which minimizes the square error loss by assigning both observed and unobserved feedback with different confidential values based on matrix factorization.
- **NGCF**, Neural Graph Collaborative Filtering [78]. This method integrates the user-item interactions into the embedding learning process, and exploits the graph structure by propagating embeddings on it to model the high-order connectivity.

We first adopt the above three backbones to learn the latent representation of users and items and obtain the relevance scores between them. Then we adopt the following three fairness-aware approaches on the top of the three backbones to achieve fair recommendation for a comparison with our proposed method.

- **FOEIR**, Fairness of Exposure in Rankings [72], which is a fairness-aware algorithm incorporating a standard linear program and the Birkhoff-von Neumann decomposition [7].
- **FairRec**, which is a two-sided fairness-aware method achieving envy-freeness up-to-one on the user side and exposure guarantee on the item side [58]. It is motivated by the fair allocation [10] and adopts the Greedy-Round-Robin algorithm [8, 15] to allocate item candidates to users.
- **TFROM**, which is a two-sided fairness-aware method ensuring individual fairness on both consumers and producers [81]. It also uses scheduling algorithm for conducting the recommendation.

Here, we summarize how the fairness criteria are defined in these fairness-aware recommendation approaches. **FOEIR** is a ranking model for search, thus it does not model consumer-sided fairness. It considers three types of fairness on the producer side: (1) Demographic Parity: $E(G_1) = E(G_2)$, which ensures that the exposure (E) obtained by different demographic groups be equal. (2) Disparate Treatment: $\frac{E(G_1)}{U(G_1)} = \frac{E(G_2)}{U(G_2)}$, which ensures that the exposure (E) should be proportional to the utility (U) of members in each group. (3) Disparate Impact: $\frac{CTR(G_1)}{U(G_1)} = \frac{CTR(G_2)}{U(G_2)}$, which cares more about the final effect and ensures that the click-through rate (CTR) of items belonging to different groups should be proportional to the utility (U). In our experiment, we choose the Demographic Parity as the fairness constraint of FOEIR. **FairRec** uses the *envy-freeness-up-to-one (EF1)* good to define the individual fairness on the consumer side: $v_{u_1}(\mathcal{A}_{u_1}) \geq v_{u_1}(\mathcal{A}_{u_2} \backslash p)$. This definition comes from the fair allocation, which is a sub-field of economics. Here $v_{u_1}(\mathcal{A}_{u_1})$ denotes the amount how u_1 values his obtained allocation (recommended items) $\mathcal{A}_{u_1}$, and p is any item in another user u_2's allocation $\mathcal{A}_{u_2}$. For the producer-sided fairness, the authors define it as ensuring a (self-defined) minimum threshold of exposure for all items. **TFROM** still defines an individual fairness on the consumer side as ensuring the NDCG values of any two users are equal: $NDCG_{u_1} = NDCG_{u_2}$. The authors define two producer-sided fairness criteria both at an individual level: (1) Uniform Fairness: $\frac{E(p_1)}{|I_{p_1}|} = \frac{E(p_2)}{|I_{p_2}|}$, which ensures that the exposure of each individual producer should be proportional to the number of items ($|I|$) she offers. (2) Quality Weighted Fairness: $\frac{E(p_1)}{Q(I_{p_1})} = \frac{E(p_2)}{Q(I_{p_2})}$, which ensures that the exposure of each individual producer should be proportional to the quality (Q) of items she offers. We summarize all these fairness criteria in Table. 5.

Lastly, we adopt our proposed **MultiFR** method on the top of the BPRMF, WRMF, and NGCF to form our final model. Our method allows the weights on different objectives to be adaptively learned during the training process with the model embeddings.

Table 5. Definitions of fairness in the three fairness-aware approaches we selected for comparison. The definitions of notations are elaborated in Section. 6.3.

Approach	Consumer-sided Fairness	Producer-sided Fairness
FOEIR [72]	N.A.	**Group Fairness** Demographic Parity: $E(G_1) = E(G_2)$ Disparate Treatment: $\frac{E(G_1)}{U(G_1)} = \frac{E(G_2)}{U(G_2)}$ Disparate Impact: $\frac{CTR(G_1)}{U(G_1)} = \frac{CTR(G_2)}{U(G_2)}$
FairRec [58]	**Individual Fairness** EF1: $v_{u_1}(\mathcal{A}_{u_1}) \geq v_{u_1}(\mathcal{A}_{u_2} \backslash p)$	**Group Fairness** Guarantee a threshold of exposure for all producers
TFROM [81]	**Individual Fairness** $NDCG_{u_1} = NDCG_{u_2}$	**Individual Fairness** Uniform Fairness: $\frac{E(p_1)}{\lvert I_{p_1} \rvert} = \frac{E(p_2)}{\lvert I_{p_2} \rvert}$ Quality Weighted Fairness: $\frac{E(p_1)}{Q(I_{p_1})} = \frac{E(p_2)}{Q(I_{p_2})}$

6.4 Experiment Settings

In the experiments, we optimize all models using the Adam optimizer with the Xavier initialization [32]. The embedding size is fixed to 50 and the batch size to 1024 for all baseline models. The learning rate and the regularization hyper-parameter are selected from $\{1e^{-1}, 1e^{-2}, 1e^{-3}, 1e^{-4}, 1e^{-5}\}$. The patience parameter γ is selected from $\{0.5, 0.6, 0.7, 0.8\}$. The smooth temperature in SmoothRank is selected from $\{1e^{-1}, 1e^{-2}, 1e^{-3}\}$. The K value in NDCG@K used for computing the consumer-side fairness described in Section 4.2.1 is set as 20. For all the datasets, we randomly sample one unobserved item as the negative sample for each user to speed up the training process. Further, for the FOEIR model, since it requires to solve a linear program with size $|\mathcal{I}| \times |\mathcal{I}|$ for each consumer with huge computational costs, we rerank the top-100 items from the base model then select the new top-K ($K < 100$) as the final recommendation. Early stopping strategy is performed, i.e., permutate stopping if Recall@20 on the validation data does not increase for 50 successive evaluation steps, for which the evaluation process is conducted for every five epochs. All experiments are conducted with PyTorch running on GPU machines (Nvidia Tesla P100).

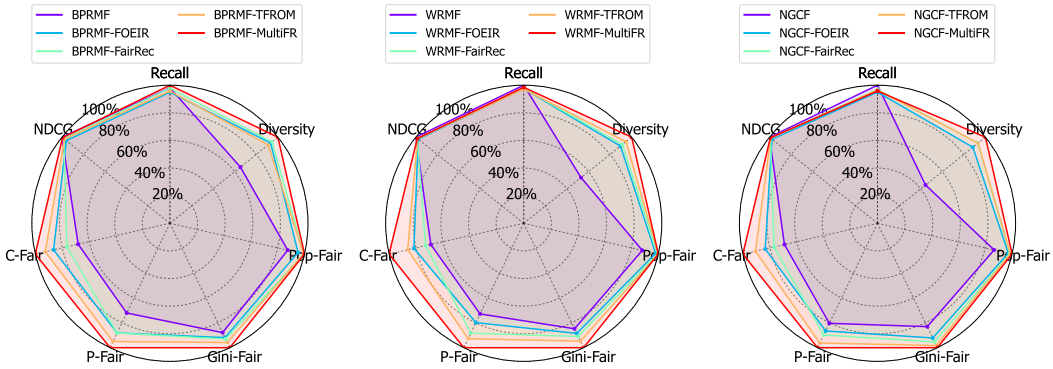
6.5 Experimental Results and Analysis

6.5.1 Overall Performance Comparison

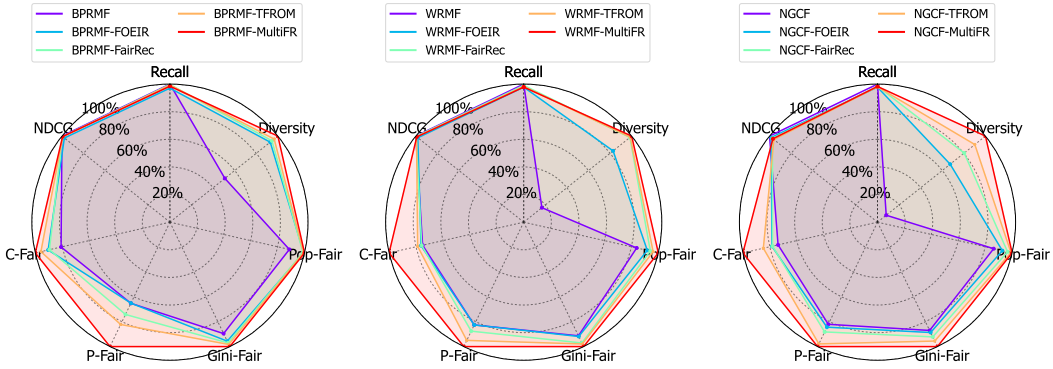
The overall experiments on three datasets are reported in Table 6, Table 7, and Table 8, respectively. In each block, bold scores are the best for each metric, while underlined scores are the second best.

Our model achieves obvious and significant improvements regarding all the fairness and diversity metrics. For instance, on the *MovieLens100K* dataset, considering the top-10 recommendation, BPRMF-MultiFR reduces the disparity on the user side by 27.27% and 32.12% on the item side compared with the BPRMF base model. WRMF-MultiFR reduces the Gini index and Popularity rate by 13.04% and 13.34%, respectively. And NGCF-MultiFR model improves the system's diversity from 0.1023 to 0.3042, which is a rather great enhancement. The biggest improvement of the diversity metric is on the *FM-reduced* dataset, where the diversity measure is improved from 0.0211 to 0.1674 by BPRMF-MultiFR compared with the corresponding base model. WRMF-MultiFR and NGCF-MultiFR also largely enhance the diversity by a large margin. Furthermore, compared with other state-of-the-art fair ranking methods, Multi-FR can still consistently achieve better fairness measures on both sides.

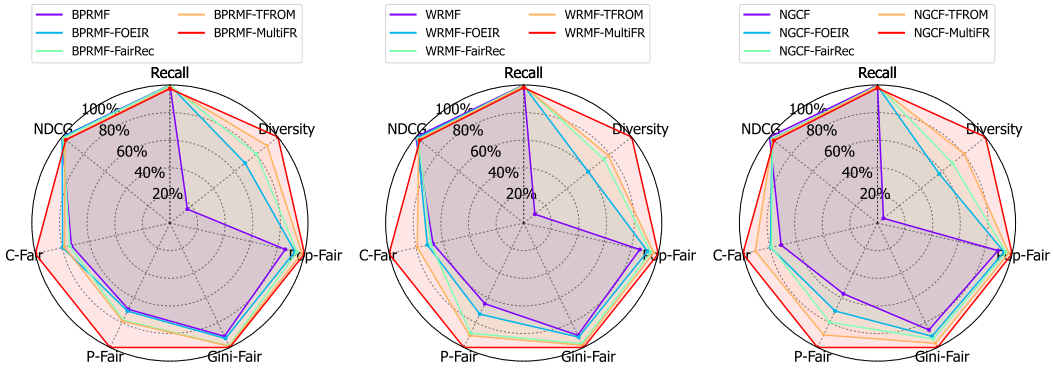
We also observe a conflict between the recommendation accuracy and fairness. For instance, NGCF achieves the highest accuracy regarding Recall and NDCG on three datasets; however,



(a) **MovieLens100K**



(b) **MovieLens1M**



(c) **FM-reduced**

Fig. 1. Relative performance achievement compared to the best on each metric given the same backbone and dataset. All the metrics are computed at position 20 (i.e., *Metric@20*). For *Recall*, *NDCG*, and *Diversity* (the larger, the better), we divide the value achieved by each method by the best one to indicate the relative performance on the recommendation accuracy of each method compared to the best one. For other metrics (the smaller, the better), we first compute the reciprocal of each value to indicate the fairness from different perspectives, followed by adopting the same operation we conduct before. Thus, the greater the percentage value in these plots, the better the approach on that measurement.

Table 6. Summary of the performance on **MovieLens100K**. We evaluate for recommendation accuracy (*Recall* and *NDCG*) and fairness (*Disparity*$_u$, *Disparity*$_i$, *Gini*, *Popularity rate*, and *Diversity*), where K is the length of the recommendation list. A metric followed by "↑" means "the larger, the better", while a metric followed by "↓" means "the smaller, the better". The fairness constraints are specified using the *"gender"* on the consumer side and the *"genre"* on the producer side. In each block, the paired t-test between the second best method and the best method on each metric is significant at $p \leq 0.01$.

Model	Recall@K ↑	NDCG@K ↑	Disparity$_u$ ↓	Disparity$_i$ ↓	Gini ↓	Popularity rate ↓	Diversity ↑
			K=10				
BPRMF	**0.2152**	**0.2637**	1.3580	1.2131	0.6919	0.8996	0.1838
BPRMF-FOEIR	0.1968	0.2473	1.2103	1.0023	0.6558	0.8351	0.2817
BPRMF-FairRec	0.2049	0.2495	1.3627	0.9357	0.6325	0.8214	0.2916
BPRMF-TFROM	0.2037	0.2481	1.1703	0.9223	0.6138	0.8172	0.2968
BPRMF-MultiFR	0.2055	0.2516	**0.9877**	**0.8235**	**0.6027**	**0.8032**	**0.3029**
WRMF	**0.2166**	**0.2748**	1.3753	1.2215	0.7278	0.9256	0.1391
WRMF-FOEIR	0.2107	0.2694	1.2533	1.1185	0.7152	0.8574	0.2478
WRMF-FairRec	0.2140	0.2735	1.3624	0.9626	0.6954	0.8315	0.2621
WRMF-TFROM	0.2128	0.2707	1.2213	0.9266	0.6424	0.8106	0.2754
WRMF-MultiFR	0.2156	0.2733	**0.9997**	**0.8672**	**0.6239**	**0.8021**	**0.3045**
NGCF	**0.2275**	**0.2855**	1.4226	1.2333	0.7526	0.9621	0.1023
NGCF-FOEIR	0.2242	0.2705	1.2624	1.1388	0.7239	0.8627	0.2282
NGCF-FairRec	0.2252	0.2776	1.3526	1.0221	0.7027	0.8410	0.2518
NGCF-TFROM	0.2219	0.2756	1.2317	1.0005	0.6811	0.8313	0.2724
NGCF-MultiFR	0.2245	0.2752	**1.0232**	**0.9862**	**0.6428**	**0.8213**	**0.3042**
			K=20				
BPRMF	**0.3273**	**0.2848**	2.8215	1.2237	0.6840	0.8630	0.2367
BPRMF-FOEIR	0.3107	0.2734	2.2250	1.0011	0.6540	0.7934	0.3381
BPRMF-FairRec	0.3204	0.2801	2.5129	1.0001	0.6459	0.7589	0.3426
BPRMF-TFROM	0.3134	0.2782	2.0781	0.9283	0.6271	0.7602	0.3397
BPRMF-MultiFR	0.3210	0.2833	**1.9234**	**0.8826**	**0.6011**	**0.7552**	**0.3625**
WRMF	**0.3305**	**0.2964**	3.0307	1.2382	0.7223	0.8953	0.1905
WRMF-FOEIR	0.3221	0.2896	2.5632	1.1296	0.6916	0.8051	0.3217
WRMF-FairRec	0.3227	0.2925	2.8691	1.0214	0.6729	0.7926	0.3281
WRMF-TFROM	0.3209	0.2912	2.4253	0.9721	0.6462	0.7891	0.3402
WRMF-MultiFR	0.3255	0.2921	**2.0913**	**0.9023**	**0.6124**	**0.7889**	**0.3588**
NGCF	**0.3431**	**0.3022**	3.2431	1.2347	0.7728	0.9233	0.1525
NGCF-FOEIR	0.3259	0.2953	2.6877	1.1465	0.6966	0.8234	0.3029
NGCF-FairRec	0.3327	0.2996	2.9162	1.1056	0.6735	0.8134	0.3194
NGCF-TFROM	0.3317	0.2989	2.4726	1.0314	0.6534	0.8108	0.3356
NGCF-MultiFR	0.3286	0.3000	**2.2421**	**0.9928**	**0.6421**	**0.8001**	**0.3429**

its recommendation is the least fair and diverse compared to other models. FOEIR, FairRec, and TFROM achieve better fairness by re-ranking the recommendation list based on the relevance scores obtained from the base models; however, the original ranking order is disrupted, leading to the accuracy drop. Different from prior post-processing methods that require to obtain the relevance scores between users and items beforehand, our Multi-FR is an in-processing method that ensures both the recommendation accuracy and multi-stakeholder fairness in an end-to-end way. Based on the experimental results, Multi-FR can balance the recommendation accuracy and fairness well by largely improving the fairness and diversity with little drop in the accuracy. For instance, concerning Recall@20, NGCF-MultiFR only has a drop of 4.23%, 1.71%, and 2.23% on three datasets, respectively, compared to the original NGCF model. Considering the large magnitude of fairness and diversity improvements, we denote this accuracy drop is relatively small.

Table 7. Summary of the performance on **MovieLens1M**. We evaluate for recommendation accuracy (*Recall* and *NDCG*) and fairness (*Disparity*$_u$, *Disparity*$_i$, *Gini*, *Popularity rate*, and *Diversity*), where K is the length of the recommendation list. A metric followed by "↑" means "the larger, the better", while a metric followed by "↓" means "the smaller, the better". The fairness constraints are specified using the *"gender"* on the consumer side and the *"genre"* on the producer side. In each block, the paired t-test between the second best method and the best method on each metric is significant at $p \leq 0.01$.

Model	Recall@K ↑	NDCG@K ↑	Disparity$_u$ ↓	Disparity$_i$ ↓	Gini ↓	Popularity rate ↓	Diversity ↑
			K=10				
BPRMF	**0.1462**	**0.2360**	1.5225	1.2648	0.7586	0.9326	0.1264
BPRMF-FOEIR	0.1425	0.2318	1.4263	1.2624	0.7171	0.8530	0.2545
BPRMF-FairRec	0.1453	0.2344	1.4927	1.0826	0.6927	0.8531	0.2637
BPRMF-TFROM	0.1437	0.2332	1.3782	0.9263	0.6918	0.8346	0.2812
BPRMF-MultiFR	0.1458	0.2333	**1.0235**	**0.8716**	**0.6825**	**0.8214**	**0.3023**
WRMF	**0.1681**	**0.2850**	2.1773	1.3125	0.7720	0.9921	0.0157
WRMF-FOEIR	0.1646	0.2809	2.1750	1.3128	0.7710	0.9244	0.0873
WRMF-FairRec	0.1661	0.2846	2.1023	1.1352	0.7241	0.9027	0.1015
WRMF-TFROM	0.1654	0.2829	1.9273	1.1044	0.7172	0.8823	0.1025
WRMF-MultiFR	0.1644	0.2811	**1.6523**	**0.9726**	**0.7032**	**0.8527**	**0.1029**
NGCF	**0.1782**	**0.2852**	2.5632	1.3527	0.8010	0.9935	0.0032
NGCF-FOEIR	0.1762	0.2834	2.3345	1.3189	0.7786	0.9305	0.0127
NGCF-FairRec	0.1774	0.2848	2.5413	1.2635	0.7309	0.9135	0.1000
NGCF-TFROM	0.1767	0.2833	2.1311	1.1472	0.7222	0.8873	0.1124
NGCF-MultiFR	0.1724	0.2829	**1.8528**	**1.1125**	**0.7152**	**0.8734**	**0.1320**
			K=20				
BPRMF	**0.2287**	**0.2438**	3.2123	1.2638	0.7512	0.9047	0.1743
BPRMF-FOEIR	0.2220	0.2384	2.8707	1.2660	0.7034	0.8075	0.3183
BPRMF-FairRec	0.2280	0.2425	2.9012	1.1109	0.6931	0.8123	0.3237
BPRMF-TFROM	0.2272	0.2419	2.7100	1.0019	0.6846	0.8087	0.3308
BPRMF-MultiFR	0.2252	0.2424	**2.5972**	**0.8241**	**0.6728**	**0.8027**	**0.3426**
WRMF	**0.2525**	**0.2859**	3.9079	1.3105	0.7579	0.9808	0.0377
WRMF-FOEIR	0.2469	0.2804	3.8470	1.3105	0.7534	0.8967	0.1854
WRMF-FairRec	0.2502	0.2851	3.8721	1.2358	0.7129	0.8749	0.2203
WRMF-TFROM	0.2482	0.2839	3.7247	1.1392	0.7072	0.8562	0.2210
WRMF-MultiFR	0.2470	0.2832	**2.9341**	**1.0826**	**0.6923**	**0.8231**	**0.2239**
NGCF	**0.2633**	**0.2936**	4.1124	1.3469	0.7992	0.9922	0.0123
NGCF-FOEIR	0.2574	0.2890	3.8728	1.3098	0.7842	0.9231	0.1026
NGCF-FairRec	0.2591	0.2856	3.8927	1.2533	0.7542	0.8862	0.1224
NGCF-TFROM	0.2580	0.2819	3.5820	1.1301	0.7278	0.8635	0.1374
NGCF-MultiFR	0.2588	0.2844	**3.0375**	**1.1057**	**0.6955**	**0.8562**	**0.1524**

In order to show the capability of our proposed method for balancing the recommendation accuracy and fairness more clearly, we display the radar plots in Fig. 1, where each sub-plot compares the FOEIR, FairRec, TFROM, and Multi-FR on a specific backbone. For Recall, NDCG, and Diversity, we divide the value achieved by each method by the highest value to indicate how much percentage different methods can reach compared to the best one given the same backbone and dataset. For other metrics which are the smaller the better, we first compute a reciprocal of those values, indicating the fairness on different perspectives, i.e., Consumer (C), Producer (P), Gini, Popularity (Pop). Then we adopt a similar way to divide the value achieved by each method by the highest value obtained across all approaches to determine the relative scale that each method may attain in comparison to the best. As illustrated in Fig. 1, the Multi-FR method outperforms all

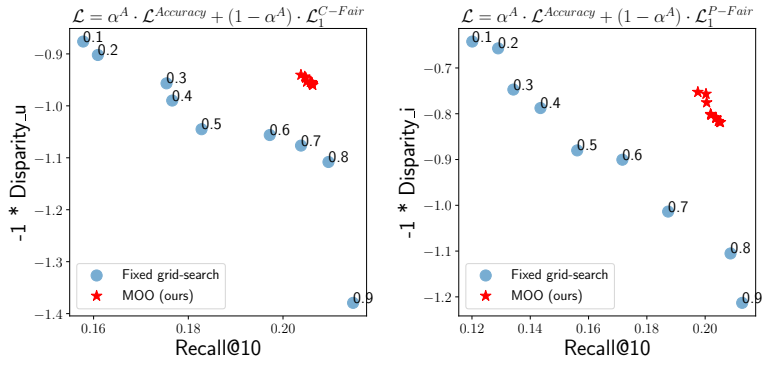
Table 8. Summary of the performance on **FM-reduced**. We evaluate for recommendation accuracy (*Recall* and *NDCG*) and fairness (*Disparity*$_u$, *Disparity*$_i$, *Gini*, *Popularity rate*, and *Diversity*), where K is the length of the recommendation list. A metric followed by "↑" means "the larger, the better", while a metric followed by "↓" means "the smaller, the better". The fairness constraints are specified using the *"age"* on the consumer side and the *"popularity"* on the producer side. In each block, the paired t-test between the second best method and the best method on each metric is significant at $p \leq 0.01$.

Model	Recall@K ↑	NDCG@K ↑	Disparity$_u$ ↓	Disparity$_i$ ↓	Gini ↓	Popularity rate ↓	Diversity ↑
			K=10				
BPRMF	**0.1248**	**0.1671**	1.3277	1.3099	0.8136	0.9893	0.0211
BPRMF-FOEIR	0.1245	0.1669	1.2746	1.2801	0.8064	0.9733	0.0732
BPRMF-FairRec	0.1240	0.1659	1.3320	1.1511	0.7826	0.9625	0.1027
BPRMF-TFROM	0.1227	0.1632	1.1371	1.1023	0.7644	0.9285	0.1266
BPRMF-MultiFR	0.1209	0.1592	**1.0288**	**0.9323**	**0.7514**	**0.9023**	**0.1674**
WRMF	**0.1326**	**0.1828**	1.6231	1.5135	0.8523	0.9905	0.0104
WRMF-FOEIR	0.1323	0.1825	1.5268	1.3687	0.8271	0.9625	0.1162
WRMF-FairRec	0.1322	0.1820	1.5826	1.2231	0.8038	0.9699	0.1008
WRMF-TFROM	0.1320	0.1817	1.3917	1.1343	0.7996	0.9555	0.1229
WRMF-MultiFR	0.1301	0.1784	**1.1273**	**1.0824**	**0.7823**	**0.9275**	**0.1462**
NGCF	**0.1452**	**0.1923**	1.8231	1.8326	0.9006	0.9932	0.0096
NGCF-FOEIR	0.1428	0.1899	1.6092	1.5247	0.8725	0.9755	0.0976
NGCF-FairRec	0.1435	0.1901	1.6235	1.3825	0.8522	0.9826	0.0927
NGCF-TFROM	0.1430	0.1892	1.4326	1.3728	0.8364	0.9678	0.1072
NGCF-MultiFR	0.1426	0.1888	**1.2526**	**1.1081**	**0.8002**	**0.9388**	**0.1388**
			K=20				
BPRMF	**0.1904**	**0.1892**	1.3658	1.3103	0.8161	0.9792	0.0407
BPRMF-FOEIR	0.1899	0.1888	1.2541	1.3746	0.8006	0.9033	0.1752
BPRMF-FairRec	0.1902	0.1872	1.3435	1.1627	0.7519	0.8892	0.2016
BPRMF-TFROM	0.1871	0.1836	1.2762	1.1517	0.7498	0.8554	0.2263
BPRMF-MultiFR	0.1853	0.1726	**0.9999**	**0.9083**	**0.7426**	**0.8388**	**0.2515**
WRMF	**0.2104**	**0.2031**	1.6127	1.6852	0.8627	0.9889	0.0214
WRMF-FOEIR	0.2096	0.2008	1.5179	1.4920	0.8489	0.9258	0.1237
WRMF-FairRec	0.2100	0.1984	1.5726	1.2338	0.8023	0.9022	0.1539
WRMF-TFROM	0.2087	0.1976	1.3744	1.2076	0.7926	0.8825	0.1627
WRMF-MultiFR	0.2062	0.1954	**1.0862**	**1.0927**	**0.7782**	**0.8526**	**0.2073**
NGCF	**0.2247**	**0.2258**	1.7923	1.9349	0.9138	0.9905	0.0102
NGCF-FOEIR	0.2229	0.2206	1.6138	1.5562	0.8623	0.9429	0.1058
NGCF-FairRec	0.2236	0.2197	1.6282	1.3791	0.8425	0.9273	0.1286
NGCF-TFROM	0.2225	0.2188	1.4131	1.2231	0.8123	0.9076	0.1486
NGCF-MultiFR	0.2197	0.2164	**1.2830**	**1.1001**	**0.7849**	**0.8862**	**0.1848**

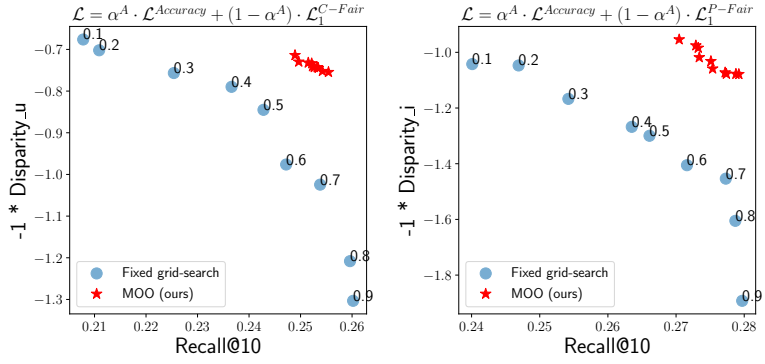
other approaches on all fairness metrics and produces nearly identical recommendation accuracy, regardless of the backbone and dataset chosen.

6.5.2 Comparison with Grid-search Strategy

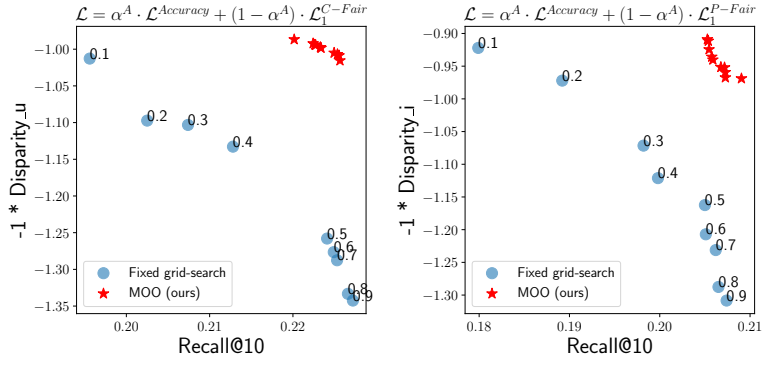
In order to demonstrate the effectiveness of the MOO mechanism in the Multi-FR, we conduct experiments to compare our model with the grid-search strategy, where scaling factors on the BPRMF objective and the fairness objective are manually set (the summation is 1). We only consider two-loss objectives for a convenient grid-search, which means we only add the fairness constraint on one side each time training with the BPR ranking loss (i.e., $\mathcal{L} = \alpha^A \cdot \mathcal{L}^{Accuracy} + (1-\alpha^A) \cdot \mathcal{L}_1^{C-Fair}$ or $\mathcal{L} = \alpha^A \cdot \mathcal{L}^{Accuracy} + (1-\alpha^A) \cdot \mathcal{L}_1^{P-Fair}$). The scatter plots are shown in Fig. 2. Each blue point indicates a grid-search solution averaged by 5 rounds where the value on the point is the weight α^A



(a) **MovieLens100K**



(b) **MovieLens1M**



(c) **FM-reduced**

Fig. 2. The comparison between the MOO mechanism in our strategy versus the grid-search strategy on all datasets with the BPRMF backbone.

on the BPR loss. Each red point refers to one final Multi-FR solution selected through the strategy described in Section 5.5 after running the model for 5 rounds. It is worthy to notice that all red points are Pareto optimal, which is theoretically guaranteed as demonstrated in Section 5. Thus,

Table 9. The training efficiency comparison of different number of fairness constraints by using our model, *Multi-FR*. The training time is reported in seconds. The backbone is chosen as the BPRMF.

Objective	*ML100K*	*ML1M*	*FM-reduced*
(1) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \alpha_1^C \cdot \mathcal{L}_1^{C-Fair}$	676.4	11,059.2	69,438.3
(2) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \alpha_1^P \cdot \mathcal{L}_1^{P-Fair}$	363.8	8,945.1	54,281.8
(3) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \sum_{i=1}^{2} \alpha_i^C \cdot \mathcal{L}_i^{C-Fair}$	749.3	15,044.0	98,177.2
(4) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \sum_{i=1}^{2} \alpha_i^P \cdot \mathcal{L}_i^{P-Fair}$	512.5	12,684.2	90,864.5
(5) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \alpha_1^C \cdot \mathcal{L}_1^{C-Fair} + \alpha_1^P \cdot \mathcal{L}_1^{P-Fair}$	912.7	19,560.7	102,232.1
(6) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \sum_{i=1}^{2} \alpha_i^C \cdot \mathcal{L}_i^{C-Fair} + \alpha_1^P \cdot \mathcal{L}_1^{P-Fair}$	1,119.2	23,653.5	123,171.7
(7) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \alpha_1^C \cdot \mathcal{L}_1^{C-Fair} + \sum_{i=1}^{2} \alpha_i^P \cdot \mathcal{L}_i^{P-Fair}$	1,013.8	23,189.3	120,816.9
(8) $\alpha^A \cdot \mathcal{L}^{Accuracy} + \sum_{i=1}^{2} \alpha_i^C \cdot \mathcal{L}_i^{C-Fair} + \sum_{i=1}^{2} \alpha_i^P \cdot \mathcal{L}_i^{P-Fair}$	1,213.5	25,793.9	165,287.6

any red point cannot dominate other red points with respect to both the recommendation accuracy and the fairness. Here, the curve of the red points can be regarded as the Pareto frontier.

From the illustration in Fig. 2, we can observe that the MOO successfully balances the trade-off between the fairness and recommendation accuracy. The clear margin distance between the curve formed by the red points (Pareto frontier) and the curve formed by the blue points show the effectiveness of the MOO mechanism in our proposed Multi-FR.

6.5.3 Training with Different Number of Constraints

We investigate the empirical training efficiency by using a different number of fairness constraints in our model. We choose BPRMF as our base model to report the training efficiency. Each row in Table 9 indicates training with a different number of disparity objectives on the consumer side and the producer side. Here the first and second fairness constraints on the consumer side represent the *gender-based fairness* and *age-based fairness*, respectively. The first and second fairness constraints on the producer side represent the *popularity-based fairness* and *genre-based fairness*, respectively. We observe that our proposed approach has reasonable training time, especially when the number of fairness constraints increases: the more number of constraints added, the less extra time the model needs. This shows the ability of our model to train multiple objectives simultaneously for multiple stakeholders in the real-world application.

7 CONCLUSION AND DISCUSSION

In this paper, we propose a multi-objective optimization framework, *Multi-FR*, for the fairness-aware recommendation in multi-sided marketplaces, where the final solution is guaranteed to be Pareto optimal. To achieve fairness-aware recommendation, four fairness constraints are proposed within the multi-objective optimization framework. We employ the smooth rank and stochastic ranking policy to make our fairness metrics differentiable, thus the fairness criteria can be optimized directly in an end-to-end way. Then, *Multi-FR* applies the multi-gradient descent algorithm to generate a Pareto set, where the scaling factors on each objectives are adaptively learned through the Frank-Wolfe Solver without handcraft tuning. Finally, the *Least Misery Strategy* is adopted to select

the most proper solution from the generated Pareto set. Experimental results on three real-world datasets show that our method can constantly outperform various corresponding base architectures and state-of-the-art fairness recommendation/ranking methods. Extensive experiments on multiple evaluation metrics clearly validate that *Multi-FR* can largely improve the recommendation fairness with only little drop in terms of the recommendation quality. Further analysis demonstrates the effectiveness of the MOO mechanism and the capability of *Multi-FR* optimizing any number of fairness criteria for multiple stakeholders concurrently.

There exist several extensions we intend to investigate as future work. Firstly, we intend to collect the producer information for the datasets, such as film producers of movies, so that we can directly define and optimize producer-sided fairness, rather than making items act as a proxy of producers. Secondly, our current definition of fairness only considers one demographic attribute at a time on the consumer side or the producer side. We intend to investigate how to ensure fairness for those people belonging to multiple demographic groups (e.g., "Black Women"). Therefore, we need to consider how to model the fairness for multiple attributes ("color" and "gender" in this example) concurrently on one side. Thirdly, the Gumbel noise used to solve the non-differential problem when modeling the exposure fairness for producers is often independently and identically sampled. This is a common practice in community. However, we would like to study whether the model uncertainty can be integrated into this sampling procedure so that the final ranking positions based on the stochastic ranking policy can be estimated with a higher quality.

ACKNOWLEDGMENTS

We thank the constructive suggestions from anonymous reviewers. This work is supported by the Microsoft Research & MILA (Quebec AI Institute) Collaboration Grant and the Start-up Grant (Grant #: 9610564) of City University of Hong Kong.

REFERENCES

[1] Himan Abdollahpouri, Robin Burke, and Bamshad Mobasher. 2017. Controlling Popularity Bias in Learning-to-Rank Recommendation. In *RecSys*. ACM, 42–46.

[2] Remzi H. Arpaci-Dusseau. 2017. Operating Systems: Three Easy Pieces. *login Usenix Mag.* 42, 1 (2017).

[3] Nina Baranchuk, Glenn MacDonald, and Jun Yang. 2011. The economics of super managers. *The Review of Financial Studies* 24, 10 (2011), 3321–3368.

[4] Ghazaleh Beigi, Ahmadreza Mosallanezhad, Ruocheng Guo, Hamidreza Alvari, Alexander Nou, and Huan Liu. 2020. Privacy-Aware Recommendation with Private-Attribute Protection using Adversarial Learning. In *WSDM*. ACM, 34–42.

[5] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H. Chi, and Cristos Goodrow. 2019. Fairness in Recommendation Ranking through Pairwise Comparisons. In *KDD*. ACM, 2212–2220.

[6] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of Attention: Amortizing Individual Fairness in Rankings. In *SIGIR*. ACM, 405–414.

[7] G. Birkhoff. 1967. *Lattice Theory* (3rd ed.). American Mathematical Society, Providence.

[8] Arpita Biswas and Siddharth Barman. 2018. Fair Division Under Cardinality Constraints. In *IJCAI*. ijcai.org, 91–97.

[9] Avishek Joey Bose and William L. Hamilton. 2019. Compositional Fairness Constraints for Graph Embeddings. In *ICML*, Vol. 97. PMLR, 715–724.

[10] Sylvain Bouveret, Yann Chevaleyre, Nicolas Maudet, and Hervé Moulin. 2016. *Fair Allocation of Indivisible Goods*. Cambridge University Press, 284–310.

[11] Sebastian Bruch, Shuguang Han, Michael Bendersky, and Marc Najork. 2020. A Stochastic Treatment of Learning to Rank Scoring Functions. In *WSDM*. ACM, 61–69.

[12] Eric Budish. 2010. The combinatorial assignment problem: approximate competitive equilibrium from equal incomes. In *BQGT*. ACM, 74:1.

[13] Robin Burke, Nasim Sonboli, and Aldo Ordonez-Gauger. 2018. Balanced neighborhoods for multi-sided fairness in recommendation. In *FAccT*. ACM, 202–214.

[14] Amelia Butterly. 2015. Google Image search for CEO has Barbie as first female result.

[15] Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. 2019. The Unreasonable Fairness of Maximum Nash Welfare. *ACM Trans. Economics and Comput.* 7, 3 (2019), 12:1–12:32.

[16] L. Elisa Celis, Damian Straszak, and Nisheeth K. Vishnoi. 2018. Ranking with Fairness Constraints. In *ICALP (LIPIcs, Vol. 107)*. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 28:1–28:15.

[17] Allison J. B. Chaney, Brandon M. Stewart, and Barbara E. Engelhardt. 2018. How Algorithmic Confounding in Recommendation Systems Increases Homogeneity and Decreases Utility. In *RecSys*. ACM, 224–232.

[18] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and Debias in Recommender System: A Survey and Future Directions. *CoRR* (2020).

[19] K. Deb, K. Sindhya, and J. Hakanen. 2016. Multi-objective optimization. In *In Decision Sciences: Theory and Practice*. CRC Press, 145–184.

[20] Jean-Antoine Désidéri. 2012. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *Comptes Rendus Mathematique* 350, 5-6 (2012), 313–318.

[21] Fernando Diaz, Bhaskar Mitra, Michael D. Ekstrand, Asia J. Biega, and Ben Carterette. 2020. Evaluating Stochastic Rankings with Expected Exposure. In *CIKM*. ACM, 275–284.

[22] Yadolah Dodge. 2008. *The Concise Encyclopedia of Statistics*. Springer New York, New York, NY, 231–233.

[23] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. 2012. Fairness through awareness. In *ITCS*. ACM, 214–226.

[24] Benjamin Edelman, Michael Luca, and Dan Svirsky. 2017. Racial Discrimination in the Sharing Economy: Evidence from a Field Experiment. *American Economic Journal: Applied Economics* 9, 2 (April 2017), 1–22.

[25] Michael D Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. 2018. All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In *FAccT*. ACM, 172–186.

[26] Michael D. Ekstrand, Mucun Tian, Mohammed R. Imran Kazi, Hoda Mehrpouyan, and Daniel Kluver. 2018. Exploring Author Gender in Book Rating and Recommendation. In *RecSys*. ACM, 242–250.

[27] Peter C. Fishburn. 1967. Letter to the Editor - Additive Utilities with Incomplete Product Sets: Application to Priorities and Assignments. *Oper. Res.* 15, 3 (1967), 537–542.

[28] Jorg Fliege and Benar Fux Svaiter. 2000. Steepest descent methods for multicriteria optimization. *Mathematical Methods of Operations Research* 51 (2000), 479–494.

[29] Marguerite Frank and Philip Wolfe. 1956. An algorithm for quadratic programming. *Naval Research Logistics Quarterly* 3, 1-2 (1956), 95–110.

[30] Yingqiang Ge, Shuchang Liu, Ruoyuan Gao, Yikun Xian, Yunqi Li, Xiangyu Zhao, Changhua Pei, Fei Sun, Junfeng Ge, Wenwu Ou, and Yongfeng Zhang. 2021. Towards Long-term Fairness in Recommendation. In *WSDM*. ACM, 445–453.

[31] A. Ghane-Kanafi and E. Khorram. 2015. A new scalarization method for finding the efficient frontier in non-convex multi-objective problems. *Applied Mathematical Modelling* 39, 23 (2015), 7483–7498.

[32] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS (JMLR Proceedings, Vol. 9)*. JMLR.org, 249–256.

[33] F. Maxwell Harper and Joseph A. Konstan. 2016. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2016), 19:1–19:19.

[34] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *ICDM*. IEEE Computer Society, 263–272.

[35] Martin Jaggi. 2013. Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization. In *ICML*, Vol. 28. PMLR, 427–435.

[36] Tamas Jambor and Jun Wang. 2010. Optimizing multiple objectives in collaborative filtering. In *RecSys*. ACM, 55–62.

[37] Matthew Kay, Cynthia Matuszek, and Sean A. Munson. 2015. Unequal Representation and Gender Stereotypes in Image Search Results for Occupations. In *CHI*. ACM, 3819–3828.

[38] I.Y. Kim and O.L. de Weck. 2006. Adaptive weighted sum method for multi-objective optimization: a new method for Pareto front generation. *Struct Multidisc Optim* 31 (2006), 105–116.

[39] Mifa Kim, Tomoyuki Hiroyasu, Mitsunori Miki, and Shinya Watanabe. 2004. SPEA2+: Improving the Performance of the Strength Pareto Evolutionary Algorithm 2. In *PPSN (Lecture Notes in Computer Science, Vol. 3242)*. Springer, 742–751.

[40] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (Aug. 2009), 30–37.

[41] H. W. Kuhn and A. W. Tucker. 1951. Nonlinear Programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, Berkeley, Calif., 481–492.

[42] David Kurokawa, Ariel D. Procaccia, and Junxing Wang. 2016. When Can the Maximin Share Guarantee Be Guaranteed?. In *AAAI*. AAAI Press, 523–529.

[43] Anja Lambrecht and Catherine Tucker. 2019. Algorithmic Bias? An Empirical Study of Apparent Gender-Based Discrimination in the Display of STEM Career Ads. *Manag. Sci.* 65, 7 (2019), 2966–2981.

[44] Xiao Lin, Hongjie Chen, Changhua Pei, Fei Sun, Xuanji Xiao, Hanxiao Sun, Yongfeng Zhang, Wenwu Ou, and Peng Jiang. 2019. A pareto-efficient algorithm for multiple objective optimization in e-commerce recommendation. In *RecSys*. ACM, 20–28.

[45] M. O. Lorenz. 1905. Methods of Measuring the Concentration of Wealth. *Publications of the American Statistical Association* 9, 70 (1905), 209–219.

[46] Chen Ma, Liheng Ma, Yingxue Zhang, Ruiming Tang, Xue Liu, and Mark Coates. 2020. Probabilistic Metric Learning with Adaptive Margin for Top-K Recommendation. In *KDD*. ACM, 1036–1044.

[47] Chen Ma, Liheng Ma, Yingxue Zhang, Haolun Wu, Xue Liu, and Mark Coates. 2021. Knowledge-Enhanced Top-K Recommendation in Poincaré Ball. In *AAAI*. AAAI Press, 4285–4293.

[48] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2016. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. In *ICLR (poster)*.

[49] Masoud Mansoury, Himan Abdollahpouri, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. 2020. FairMatch: A Graph-based Approach for Improving Aggregate Diversity in Recommender Systems. In *UMAP*. ACM, 154–162.

[50] Vijay K. Mathur. 1991. How Well Do We Know Pareto Optimality? *The Journal of Economic Education* (1991).

[51] Sean M. McNee, John Riedl, and Joseph A. Konstan. 2006. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *CHI Extended Abstracts*. ACM, 1097–1101.

[52] Rishabh Mehrotra, Ashton Anderson, Fernando Diaz, Amit Sharma, Hanna M. Wallach, and Emine Yilmaz. 2017. Auditing Search Engines for Differential Satisfaction Across Demographics. In *WWW (Companion Volume)*. ACM, 626–633.

[53] 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*. ACM, 2243–2251.

[54] Alistair Moffat and Justin Zobel. 2008. Rank-Biased Precision for Measurement of Retrieval Effectiveness. *ACM Trans. Inf. Syst.* 27, 1, Article 2 (Dec. 2008), 27 pages.

[55] Nicola Neophytou, Bhaskar Mitra, and Catherine Stinson. 2022. Revisiting Popularity and Demographic Biases in Recommender Evaluation and Effectiveness. In *ECIR (1) (Lecture Notes in Computer Science, Vol. 13185)*. Springer, 641–654.

[56] Xia Ning, Christian Desrosiers, and George Karypis. 2015. A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In *Recommender Systems Handbook*. Springer, 37–76.

[57] Xia Ning and George Karypis. 2011. SLIM: Sparse Linear Methods for Top-N Recommender Systems. In *ICDM*. IEEE Computer Society, 497–506.

[58] Gourab K. Patro, Arpita Biswas, Niloy Ganguly, Krishna P. Gummadi, and Abhijnan Chakraborty. 2020. FairRec: Two-Sided Fairness for Personalized Recommendations in Two-Sided Platforms. In *WWW*. ACM / IW3C2, 1194–1204.

[59] Gourab K. Patro, Abhijnan Chakraborty, Niloy Ganguly, and Krishna P. Gummadi. 2020. Incremental Fairness in Two-Sided Market Platforms: On Updating Recommendations Fairly. In *AAAI*. AAAI Press.

[60] Toon De Pessemier, Simon Dooms, and Luc Martens. 2014. Comparison of group recommendation algorithms. *Multim. Tools Appl.* 72, 3 (2014), 2497–2541.

[61] R. Plackett. 1975. The Analysis of Permutations.

[62] Tao Qin, Tie-Yan Liu, and Hang Li. 2010. A general approximation framework for direct optimization of information retrieval measures. *Inf. Retr.* 13, 4 (2010), 375–397.

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

[64] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*. AUAI Press, 452–461.

[65] Marco Tulio Ribeiro, Anisio Lacerda, Adriano Veloso, and Nivio Ziviani. 2012. Pareto-Efficient Hybridization for Multi-Objective Recommender Systems. In *RecSys*. ACM, 19–26.

[66] Marco Tulio Ribeiro, Nivio Ziviani, Edleno Silva De Moura, Itamar Hata, Anisio Lacerda, and Adriano Veloso. 2015. Multiobjective Pareto-Efficient Approaches for Recommender Systems. *ACM TIST* 5, 4, Article 53 (2015), 20 pages.

[67] S. Schäffler, R. Schultz, and K. Weinzierl. 2002. Stochastic Method for the Solution of Unconstrained Vector Optimization Problems. *Journal of Optimization Theory and Applications* 114 (2002), 209–222.

[68] Amartya Sen. 1973. *On Economic Inequality*. Clarendon Press, Oxford.

[69] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. In *NeurIPS*. 525–536.

[70] Dimitris Serbos, Shuyao Qi, Nikos Mamoulis, Evaggelia Pitoura, and Panayiotis Tsaparas. 2017. Fairness in Package-to-Group Recommendations. In *WWW*. ACM, 371–379.

[71] E.H. Simpson. 1949. *Measurement of Diversity*. Nature, 688–688.

[72] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of Exposure in Rankings. In *KDD*. ACM, 2219–2228.

[73] Ashudeep Singh and Thorsten Joachims. 2019. Policy Learning for Fairness in Ranking. In *NeurIPS*. 5427–5437.

[74] Hugo Steinhaus. 1969. *Mathematical snapshots*.

[75] Tom Sühr, Asia J Biega, Meike Zehlike, Krishna P Gummadi, and Abhijnan Chakraborty. 2019. Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In *KDD*. ACM, 3082–3092.

[76] Jianing Sun, Yingxue Zhang, Wei Guo, Huifeng Guo, Ruiming Tang, Xiuqiang He, Chen Ma, and Mark Coates. 2020. Neighbor Interaction Aware Graph Convolution Networks for Recommendation. In *SIGIR*. ACM, 1289–1298.

[77] Saul Vargas and Pablo Castells. 2014. Improving sales diversity by recommending users to items. In *RecSys*. ACM, 145–152.

[78] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*. ACM, 165–174.

[79] Le Wu, Lei Chen, Pengyang Shao, Richang Hong, Xiting Wang, and Meng Wang. 2021. Learning Fair Representations for Recommendation: A Graph-based Perspective. In *WWW*. ACM / IW3C2, 2198–2208.

[80] Mingrui Wu, Yi Chang, Zhaohui Zheng, and Hongyuan Zha. 2009. Smoothing DCG for learning to rank: a novel approach using smoothed hinge functions. In *CIKM*. ACM, 1923–1926.

[81] Yao Wu, Jian Cao, Guandong Xu, and Yudong Tan. 2021. TFROM: A Two-sided Fairness-Aware Recommendation Model for Both Customers and Providers. In *SIGIR*. ACM, 1013–1022.

[82] Lin Xiao, Zhang Min, Zhang Yongfeng, Gu Zhaoquan, Liu Yiqun, and Ma Shaoping. 2017. Fairness-Aware Group Recommendation with Pareto-Efficiency. In *RecSys*. ACM, 107–115.

[83] Xin-She Yang. 2014. Nature-Inspired Optimization Algorithms. In *Nature-Inspired Optimization Algorithms*. Elsevier.

[84] Sirui Yao and Bert Huang. 2017. Beyond Parity: Fairness Objectives for Collaborative Filtering. In *NeurIPS*. 2925–2934.

[85] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. FA*IR: A Fair Top-k Ranking Algorithm. In *CIKM*. ACM, 1569–1578.

[86] Jianghong Zhou, Eugene Agichtein, and Surya Kallumadi. 2020. Diversifying Multi-aspect Search Results Using Simpson's Diversity Index. In *CIKM*. ACM, 2345–2348.

[87] Ziwei Zhu, Xia Hu, and James Caverlee. 2018. Fairness-Aware Tensor-Based Recommendation. In *CIKM*. ACM, 1153–1162.