Rectifying Unfairness in Recommendation Feedback Loop

Mengyue Yang *
University College London
London, United Kingdom
mengyue.yang.20@ucl.ac.uk

Jun Wang
University College London
London, United Kingdom
jun.wang@cs.ucl.ac.uk

Jean-Francois Ton
ByteDance Research
London, United Kingdom
jeanfrancois@bytedance.com

ABSTRACT

The issue of fairness in recommendation systems has recently become a matter of growing concern for both the academic and industrial sectors due to the potential for bias in machine learning models. One such bias is that of feedback loops, where the collection of data from an unfair online system hinders the accurate evaluation of the relevance scores between users and items. Given that recommendation systems often recommend popular content and vendors, the underlying relevance scores between users and items may not be accurately represented in the training data. Hence, this creates a feedback loop in which the user is not longer recommended based on their true relevance score but instead based on biased training data. To address this problem of feedback loops, we propose a two-stage representation learning framework, B-FAIR, aimed at rectifying the unfairness caused by biased historical data in recommendation systems. The framework disentangles the context data into sensitive and non-sensitive components using a variational autoencoder and then applies a novel Balanced Fairness Objective (BFO) to remove bias in the observational data when training a recommendation model. The efficacy of B-FAIR is demonstrated through experiments on both synthetic and real-world benchmarks, showing improved performance over state-of-the-art algorithms.

CCS CONCEPTS

• **Information systems** → **Information systems applications**; **Web mining**.

KEYWORDS

fairness learning, recommendation system, recommendation feedback loop, user-item fairness, unbias learning

ACM Reference Format:
Mengyue Yang, Jun Wang, and Jean-Francois Ton . 2023. Rectifying Unfairness in Recommendation Feedback Loop. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3539618.3591754

1 INTRODUCTION

Popular online recommendation platforms such as Amazon, Yelp, TikTok etc, aim to help customers browse items online in an efficient manner by recommending the user personalized items that they might be interested in. By providing a recommendation policy, these platforms can connect customers and item producers by training their models on a huge corpus of self-collected logged data. However, recently the question of social responsibility in the recommendation systems has attracted a lot of attention. In particular, users are wondering how the recommendation system works and whether there are any biases in these systems [5, 30].

Consider the advertisement industry in the cosmetics sector as an example. The logged training data in this scenario is likely to contain biases that result in the overwhelming recommendation of makeup products to women. However, some women may not have an interest in cosmetics. This phenomenon is due to unfair feedback loops, where the correlation between women and makeup is being amplified at each recommendation iteration. While gender is considered as sensitive information, and may not be explicitly used as an input to the recommendation system, it can be inferred from other inputs that are correlated with gender, such as purchasing history or social media following. Additionally, there are other biases such as item-specific biases in recommendation systems, where popular items tend to receive more exposure compared to less popular ones, limiting the audience for smaller vendors [36, 45].

The unfairness problem from training recommendation models on biased logged data is well-known [24, 30]. Building a model based on data that was generated by one's own recommendation systems will inevitably create a feedback loop which amplify biases [9, 17] and where the true relevance score is harder to determine.

Traditionally, most existing methods consider the approach of designing fairness constraints based on sensitive information to guarantee groups or individuals are treated fairly under the online policy [20–22, 25]. These model-based approaches mainly focus on fair predictions, i.e., exposure being independent of the sensitive attributes, which can degrade the performance of the original recommendation model [24, 39].

Contrary to previous methods, we tackle the fairness problem from a data debiasing perspective. Specifically, we focus on learning debiased representations of the data, which can then be used as inputs for any downstream recommendation models. This can be seen as a fair feature extraction step before applying the actual recommendation model. The main goal in our case is to determine the unbiased true relevance score of the user-item pair and thus achieve fairness by unbiasedly recommending items to the user.

Hence, this approach from a debiasing perspective is orthogonal to standard fairness constraint-based methods, as both of these approaches can be combined [11, 12]. We leave this combination of literature for future work as it is out of the scope of this paper.

More precisely, in this work we introduce a new concept of fairness called *balanced fairness* from a data debiasing perspective. We argue that a model is *balanced fair* and does not suffer from feedback

*This work is done during the first author's internship in ByteDance Research.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan

ACM ISBN 978-1-4503-9408-6/23/07.
https://doi.org/10.1145/3539618.3591754

loops, if it has been trained on a dataset, in which, recommendations were selected uniformly based on sensitive attributes (further discussed in Section 3). In this scenario, we show empirically (Section 5) that there is no unfair feedback loop that reinforces biases, as the model is being trained unbiasedly at each time step. This enables us to calculate the true relevance score between user and item. However, in reality such training data is not readily available due to practical constraints i.e. users leaving because recommendations are uniformly at random and not personalized [35].

Hence, to achieve *balanced fairness* we develop a two-stage representation learning framework as follows: Firstly, given context features (i.e. input to the model which can come from both users and items), we extract the sensitive-correlated information into representations using an identifiable VAE [16]. Secondly, we learn a second-level representation of these sensitive representations to remove biases across sensitive groups by proposing a new adversarial learning strategy. Lastly, these representations are used as input to any recommendation model.

We show that by adopting our debiasing training framework, we can train a recommendation model *as if the data came from a balanced/unbiased dataset*, while only having access to biased data. In addition, given that we extract the sensitive-correlated information into representations in the first stage, we can also check how much of the sensitive information has been removed (See Ablation study in section 5.4).

The main contribution of this paper are summarized below:

- We propose a new type of fairness from a of data debiasing perspective, which we term *balanced fairness* and develop an objective called Balanced Fairness Objective (BFO).
- Next we present a two-stage end-to-end algorithm (B-FAIR), which given biased unfair logged training data allows us to train representations for any recommendation model as if the data came from an unbiased and fair dataset.
- Lastly, we show the effectiveness of our method B-FAIR over existing methods in synthetic as well as real-world experiments.

2 RELATED WORKS

Fairness has recently attracted a lot of attention in recommendation system community [37, 44]. The key objective in fairness is that groups or individuals should be treated independently of their sensitive attributes such as gender etc. For clarity, we introduce these stages separately in section 2.1 and 2.2 respectively.

2.1 Debiasing recommendation

Given that unfairness can be caused by data bias [12], the debiasing objective can be reduced to solving popularity bias and exposure bias induced by a missing-not-at-random problem in the dataset [7, 8, 34, 46]. In particular for recommendation systems, the lack of interaction between the user and the item does not necessarily signify that the user was not interested. This missing interaction in the data could be due to the recommendation model not exposing the user to the item and hence computing the true user-item relevance score becomes increasingly hard. Recommendation models trained on such data could be heavily biased and thus reinforce unfairness. There are three types of methods for data debiasing (1) **Sample level objective**: Sample level methods aim to simulate the user's preferences for unexposed items such that the model can be

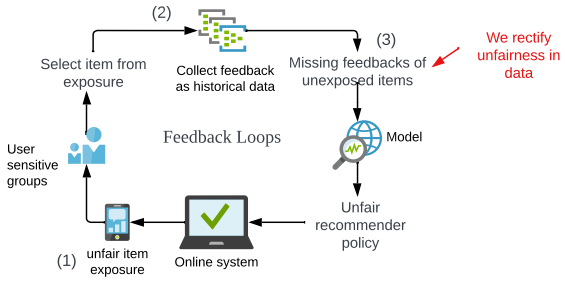


Figure 1: The figure illustrates the unfairness caused by feedback loops in recommendation systems.

trained directly on the entire sample space, where every user-item interaction has been "*observed*" [19]. (2) **Optimization level objective**: These methods are based on distribution adjustments, where importance sampling and re-weighting of the data are applied to optimization objectives [4, 19, 29, 30, 32, 33]. (3) **Representation learning**: Lastly, representation learning methods [40] aim to learn mappings of the data such that the representation is not affected by the biased historical dataset. Our method B-FAIR is most closely related to the representation learning perspective and contrary to the existing debiasing method, B-FAIR focuses on eliminating the unfairness in training data stemming from sensitive information.

2.2 Fairness recommendation method

For the algorithmic level objective, previous methods mostly focus on how to achieve fair policies by exposing similar items across sensitive groups in an online recommendation system. The key ingredient in most of these methods is to develop fairness constraints, which allows them to obtain fair item exposure, i.e. exposure to items independent of sensitive attributes such as gender [1, 6, 13, 26]. In addition, some works consider a re-ranking/post-processing step of the outputs of recommendation systems to increase fairness [3, 20, 38, 43]. In summary, previous fairness approaches to recommendation systems are mostly concerned with learning a model with fairness constraints i.e. predictions being independent of the sensitive attributes. However, in this paper, we focus on tackling the unfairness created by the feedback loop, i.e. biased logged data and hence we will concentrate on the second fairness objective proposed in [11], which is that of data debiasing fairness. To reiterate, both of these objectives mentioned in [11] are orthogonal to each other and hence could be used in conjunction.

3 PROBLEM DEFINITION AND SETUP

3.1 Notation and setup

We denote $\mathbf{c} \in \mathbb{R}^d$ as a context variable which is collected from an online recommendation system, such as the user and item context features e.g. user profile and item attributes. $i \in [1, \cdots, m]$ denotes the item index from the vendor and $o_i \in \{0, 1\}$ indicates if the item i is exposed to the user. $y \in \{0, 1\}$ denotes the feedback of user u for item i. $g_u \in \mathcal{G}_u$ and $g_i \in \mathcal{G}_i$ denote the sensitive group features on the user side and producer side respectively. For the recommendation setting, g_u corresponds to the indicator variable for sensitive information such as gender and g_i corresponds to sensitive information for items such as movie tags in the movie

recommendation system. Given that we are using a representation learning framework, we denote $z_u \in \mathbb{R}^{l_u}$, $z_i \in \mathbb{R}^{l_i}$ and $z_n \in \mathbb{R}^{l_n}$ as neural network representations of the user sensitive features (i.e. the features correspond to the sensitive information lile gender), item sensitive features, and other non-sentive features respectively. We will go into much more detail about these representations in section 4. Finally, we denote $\mathbb{D} = \left\{\left(\mathbf{c}^n, o_i^n, {g_u}^n, {g_i}^n, y^n\right)\right\}_{n=1}^{N}$ as the logged training dataset, which has been collected in a biased and unfair fashion.

3.2 Fair exposure in feedback loops

Before diving deeper into our formulation of *balanced fairness*, we would like to clarify the fairness that we aim to tackle and remove. To this end, we illustrate the feedback loop in figure 1 to shed more light on the problem. In particular, figure 1 depicts the relationship among the components in the recommendation pipeline.

- Firstly, the users are exposed to a selection of items recommended by a recommendation policy.
- These interactions are then recorded and generate the data that is stored as historical data.
- Finally, this stored data, which might include biases based on sensitive attributes, is used to train the recommendation policy used in step 1.

The key problem is that the recommendation policy might not extract the true relevance score for a user-item pair but rather a pattern from historical data, which is being reinforced with every iteration [2, 29, 30]. Going back to our leading example, if the logged data has abundant examples of female users interacting with cosmetic products, the policy might wrongly recommend items to female users who are not interested in these products.

Hence, aiming to find the true personalized relevance score between the user-item is crucial to ensure balanced and fair recommendations and is the core question that we aim to tackle in this paper. One way to better estimate the true user-item relevance is to collect the training data from the online system with a uniform item exposure probability conditioned on the sensitive group. The uniform/fair exposure in historical data is formally defined below.

Definition 1. (Fair Exposure) Assuming that there is uniform exposure conditioned on the user-sensitive features $\mathbf{z_u}$ and item-sensitive features $\mathbf{z_i}$, the following two equations hold:

- $p(o_{i=1}|\mathbf{z_u}, \mathbf{z_i}) = \cdots = p(o_{i=m}|\mathbf{z_u}, \mathbf{z_i}) = p^{uni}(o_i|\mathbf{z_u}, \mathbf{z_i}) = p^{uni}(o_i)$,
- $p(\mathbf{z_u}, \mathbf{z_i}|o_i = 0) = p(\mathbf{z_u}, \mathbf{z_i}|o_i = 1) = p(\mathbf{z_u}, \mathbf{z_i})$

where p^{uni} denotes uniform exposure for items conditioned on the sensitive information, i.e. $p^{uni}(o_i = 1|\mathbf{z_u}, \mathbf{z_i}) = p^{uni}(o_i = 1) = \frac{1}{2}$.

Intuitively, the idea of "*Fair Exposure*" is that being exposed to an item should not be dependent on a person's sensitive attributes, like gender or race. The ideal scenario is when the exposure to items is uniformly at random and unbiased, allowing us to accurately capture the user's feedback, or relevance score.

However, collecting data in this manner is not practical as users may leave the platform before providing accurate feedback due to biased recommendations. To address this, we propose the B-FAIR algorithm in section 4 without access to unbiased data. Note that the two equations in definition 1 infer the same independence statement because $p(\mathbf{z_u}, \mathbf{z_i}, o_i) = p(o_i)p(\mathbf{z_u}, \mathbf{z_i}|o_i) = p(\mathbf{z_u}, \mathbf{z_i})p(o_i|\mathbf{z_u}, \mathbf{z_i}) = p(\mathbf{z_u}, \mathbf{z_i})p(o_i)$.

3.3 Balanced fairness objective

In this section, we introduce our solution to the problem of unfair feedback loop in recommendation systems by proposing a novel objective, called the Balanced Fairness Objective (BFO). This objective differs from the traditional fairness metrics of demographic parity, equalized odds, and equal opportunity as follows:

Definition 2. (Balanced Fair Objective), For any loss function δ, the balanced fair objective on any downstream recommendation model f is defined as:

$$\begin{aligned} L_{\text{b}}^{f} &= \mathbb{E}_{\mathbf{c},\mathbf{z_u},\mathbf{z_i},\mathbf{z_n}} \mathbb{E}_{o_i \sim p^{\text{uni}}(o_i|\mathbf{z_u},\mathbf{z_i})} o_i[\delta(y, f(\mathbf{c}, \mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}))]. \\ &= \mathbb{E}_{\mathbf{c},o_i,\mathbf{z_n}} \mathbb{E}_{z_u,z_i \sim p(z_u,z_i)} o_i[\delta(y, f(\mathbf{c}, \mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}))]. \end{aligned} \tag{1}$$

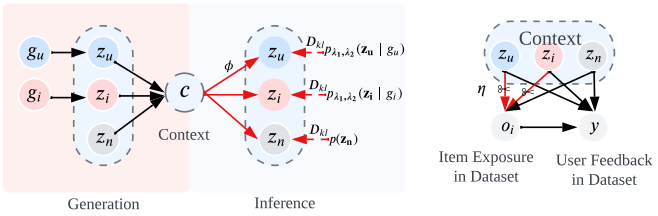
There are three key takeaways from this objective:

- Training a recommendation model under an unfair and biased empirical distribution of the exposures $\hat{p}(o_i|\mathbf{z}_u, \mathbf{z}_i)$ instead of $p^{\text{uni}}(o_i|\mathbf{z}_u, \mathbf{z}_i)$ would inevitably result in biased relevance scores due the feedback loop [17, 30], i.e. unfair up or down weighting for specific items. Hence, by optimizing BFO, we are in fact able to estimate the true relevance score for a given user-item pair, because each item was exposed to the user uniformly at random given the sensitive attributes.
- The BFO can be easily estimated when "*Fair Exposure*" data is provided i.e. the item exposure is uniformly sampled conditioned on the sensitive attributes.
- However, as mentioned above, in the real world we rarely have access to this type of fair exposure data and hence one way to still use this objective would be to use importance sampling [10, 32, 33]. These estimators, however usually come with high variance and hence we propose a representation learning-based method in the next section, which avoids the drawbacks of importance sampling. The key idea is to remove the causal relationship between sensitive attributes and item exposure in historical data.(see fig. 2(b))

Definition 3. (Balanced Fairness) A recommendation model f^* is called *balanced fair* if the model minimizes the BFO i.e.

$$\begin{aligned} f^* &= \arg\min_f \mathbb{E}_{\mathbf{c},\mathbf{z_u},\mathbf{z_i},\mathbf{z_n}} \mathbb{E}_{o_i \sim p^{\text{uni}}(o_i|\mathbf{z}_u,\mathbf{z}_i)} o_i[\delta(y, f(\mathbf{c}, \mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}))] \\ &= \arg\min_f \mathbb{E}_{\mathbf{c},o_i,\mathbf{z_n}} \mathbb{E}_{z_u,z_i \sim p(z_u,z_i)} o_i[\delta(y, f(\mathbf{c}, \mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}))] \\ &= \arg\min_f L_{\text{b}}^{f}. \end{aligned}$$

Recall, this is different to standard fairness definitions [11, 21, 23, 31] as we are motivated from a data debiasing perspective rather than a model prediction perspective. In particular, we define *balanced fairness* through being able to train a recommendation model on fairly exposed data and thus recover the true relevance scores between user and item. Given that we do this at each iteration of the loop, we are argue that we construct a recommendation system that does not suffer from the unfairness of feedback loops. Note that, traditional fairness constraints could be added to our training framework in future work.



(a) Context generation and feature inference (b) Balanced feature learning

Figure 2: (a)The figure shows the context generation process, where ϕ is the feature inference function in the disentanglement method. (b) The main objective is to remove the red lines, i.e. removing the bias stemming from the sensitive features and item exposure using a learned representation η.

To summarize, we defined the concept of "*Fair Exposure*" and the balanced fairness objective (BFO). BFO aims to emulate the scenario where there is no unfair feedback loop present and the true relevance score between user and item can be obtained. However, computing this objective is difficult when data with fair exposure is not available. To overcome this challenge, we propose a two-stage approach based on representation learning in the next section.

4 METHOD

In this section, we describe how our proposed framework Balanced and FAIr Representations (B-FAIR) can optimize the objective given in Eq. 1, when we do not have access to unbiased data. B-FAIR is divided into two stages: First, we use an injective function $\boldsymbol{\phi} : \mathbb{R}^d \to \mathbb{R}^z$ to disentangle the sensitive and non-sensitive information into a representation $\mathbf{z} \in \mathbb{R}^z$ from context $\mathbf{c}$. The disentangled representation $\mathbf{z}$ is defined as $\mathbf{z} = [\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}]$, where $z_u \in \mathbb{R}^{l_u}$ and $z_i \in \mathbb{R}^{l_i}$ are user and item sensitive representations respectively. $\mathbf{z_n} \in \mathbb{R}^{l_n}$ captures the remaining information from context which is orthogonal to the sensitive attributes. (See figure 2(a)).

Next, these disentangled representations are fed into a balanced representation function $\eta : \mathbb{R}^{l_u+l_i} \to \mathbb{R}^b$, which we properly define in Section 4.2 to generate a fair and balanced representation with respect to the sensitive attributes. These newly learned representations, together with the non sensitive representation z_n can be used to predict the user feedback y using on any downstream recommendation model $f : \mathbb{R}^{b+l_n} \to [0, 1]$. (See fig 2(b))

4.1 Disentangle sensitive features from context

It is common in the fairness literature to assume access to sensitive attributes g_u, g_i in the dataset [21]. If we were to simply remove the sensitive attributes from the training data, we will inevitably ignore the sensitive correlated information (e.g. the occupation is correlated with age) in context data. Instead of using the group index directly, we determine the boundary between sensitive and non-sensitive information before we learn the fairness representation. To achieve this goal, we assume that the context $\mathbf{c}$ is generated by three factors $\mathbf{z} = [\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}]$, i.e. user sensitive, item sensitive and non sensitive information, respectively (see figure 2(a)). For example, if we had a sensitive attribute g_u as gender, then $\mathbf{z_u}$ would contain all the gender-correlated information from the context c. To accurately identify sensitive correlated information, we build upon the work in [16] and use a conditional variational inference approach, where we infer the hidden $\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}$ in an identifiable manner. Assuming that c is conditionally independent of g_i, g_u given z, we can write out the following generative model.

$$p_{\boldsymbol{\theta}_\mathbf{h},\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c}, \mathbf{z} \mid g_u, g_i) = p_{\boldsymbol{\theta}_h}(\mathbf{c} \mid \mathbf{z}) p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z} \mid g_u, g_i) \tag{2}$$

where $\boldsymbol{\theta}_\mathbf{h}$ are the parameters for the injective generator $\mathbf{h} : \mathbb{R}^z \to \mathbb{R}^d$ of the VAE and $\boldsymbol{\lambda_1}(x) = x \cdot \mathbf{1}, \boldsymbol{\lambda_2} = \mathbf{1}$ and $\mathbf{1}$ is a vector of ones.

The generative model $p_{\boldsymbol{\theta}_\mathbf{h},\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c}, \mathbf{z} \mid \mathbf{g_u}, \mathbf{g_i})$ can thus be decomposed into the context generation model $p_{\boldsymbol{\theta}_\mathbf{h}}(\mathbf{c} \mid \mathbf{z})$ and the conditional sensitive feature $p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z} \mid \mathbf{g_u}, \mathbf{g_i})$. We can further decompose the $p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z} \mid \mathbf{g_u}, \mathbf{g_i})$ as follows:

$$p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z} \mid g_u, g_i) = p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_u} \mid g_u) p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_i} \mid g_i) p(\mathbf{z_n}), \tag{3}$$

Intuitively, this corresponds to the independence between the z variables as shown in figure 2(a). Inspired by the previous identifiable VAEs framework [16], we design conditional multivariate Gaussian distribution for each of the components as follows:

$$\begin{aligned} p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_u} \mid g_u) &= \mathcal{N}\left(\boldsymbol{\lambda}_1(g_u), \boldsymbol{\lambda}_2(g_u)\right) \\ p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_i} \mid g_i) &= \mathcal{N}\left(\boldsymbol{\lambda}_1(g_i), \boldsymbol{\lambda}_2(g_i)\right) \\ p(\mathbf{z_n}) &= \mathcal{N}\left(\mathbf{0}, \mathbf{I}_{z_3}\right), \end{aligned} \tag{4}$$

where $\mathbf{I}_{z_3}$ a diagonal matrix of dimension z_3.

According to [16, 42], by training a VAE using the above conditional priors, we can learn an identifiable latent representation of the context $\mathbf{c}$ in terms of $\mathbf{z} = [\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}]$, where all the sensitive information of g_u is captured in $\mathbf{z_u}$. Similarly for g_i and z_i. As is standard in VAE, the learning objective is to maximize the data likelihood $\mathbb{E} \log p_{\boldsymbol{\theta}_h,\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c}, \mathbf{z} \mid g_u, g_i)$ which in general is not analytically tractable. Hence, we instead maximise the lower bound of the data likelihood which is also known as the Evidence Lower Bound (ELBO). Denoting $\mathcal{D}_{KL}(\cdot\|\cdot)$ as the KL divergence, $\mathbb{C} = \{(\mathbf{c}^n, {g_u}^n, {g_i}^n)\}_{n=1}^N$ as the observed context dataset without the user feedback Y and item exposure o_i, the learning objective can be written as,

$$\begin{aligned} &\mathbb{E}_{q_\mathbb{C}}\left[\log p_{\boldsymbol{\theta}_h,\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c}, \mathbf{z} \mid g_u, g_i)\right] \geq \text{ELBO} \\ =&\mathbb{E}_{q_\mathbb{C}}[\mathbb{E}_{q_{\boldsymbol{\theta}_\phi}(\mathbf{z}|\mathbf{c},g_u,g_i)}[\log p_{\boldsymbol{\theta}_h,\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c} \mid \mathbf{z})] \\ -&\mathcal{D}_{KL}(q_{\boldsymbol{\theta}_\phi}(\mathbf{z} \mid \mathbf{c}, g_u, g_i) \| p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z} \mid g_u, g_i))] \end{aligned} \tag{5}$$

where $\boldsymbol{\phi}$ is the feature inference function formally defined in Section 4. Due to the factorization that we described previously in Eq. 3, the $\mathcal{D}_{KL}$ term can be further decomposed further as:

$$\begin{aligned} \text{ELBO} = \mathbb{E}_{q_\mathbb{C}}[&\mathbb{E}_{q_{\boldsymbol{\theta}_\phi}(\mathbf{z}|\mathbf{c},g_u,g_i)}[\log p_{\boldsymbol{\theta}_h,\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{c} \mid \mathbf{z})] \\ &- \mathcal{D}_{kl}\left(q_{\boldsymbol{\theta}_\phi}(\mathbf{z_u} \mid \mathbf{c}, g_u, g_i) \| p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_u} \mid g_u)\right) \\ &- \mathcal{D}_{kl}\left(q_{\boldsymbol{\theta}_\phi}(\mathbf{z_i} \mid \mathbf{c}, g_u, g_i) \| p_{\boldsymbol{\lambda}_1,\boldsymbol{\lambda}_2}(\mathbf{z_i} \mid g_i)\right) \\ &- \mathcal{D}_{kl}\left(q_{\boldsymbol{\theta}_\phi}(\mathbf{z_n} \mid \mathbf{c}, g_u, g_i) \| p(\mathbf{z_n})\right)] \end{aligned} \tag{6}$$

The details on the derivation of this ELBO are given in the section 6.2. With this form, we can implement a loss function to train the disentanglement part of our method. Since the prior in the final term is a standard multivariate Gaussian distribution and is a spherical Gaussian [16], we can not guarantee the identifiability of $\mathbf{z_n}$. Therefore, we add a constraint to the final objective

$$L_n = \mathcal{D}_{kl}(q_{\boldsymbol{\theta}_\phi}(\mathbf{z_n} \mid \mathbf{c}, g_u, g_i) \| \mathcal{N}\left(\boldsymbol{\lambda}_1(g_u, g_i), \boldsymbol{\lambda}_2(g_u, g_i)\right))$$

to make sure $\mathbf{z_n}$ retains the information from context and removes all the sensitive correlated information. The final disentangling objective can thus be written as below, where λ is a hyperparameter.

$$L_{dis}(\theta_{\boldsymbol{\phi}}, \theta_{\mathbf{h}}) = -ELBO - \lambda L_n \tag{7}$$

By optimizing $L_{dis}(\theta_{\boldsymbol{\phi}}, \theta_{\mathbf{h}})$, we are able to learn a disentangled representation $\mathbf{z} = [\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}]$ of the context $\mathbf{c}$.

In summary, we use an iVAE [16] with a specific prior on the latent space z to obtain a disentangled representation $\mathbf{z} = [\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}]$ of the context c in terms of sensitive and non-sensitive attributes. With these disentangled representations, we move on to the second stage, which takes $\mathbf{z_u}, \mathbf{z_i}$ (user and item sensitive feature representations) as input to learn fair balanced recommendation policy, i.e. learning a model which breaks the feedback loop by training under the Balanced Fair Objective (BFO) in Eq. 1.

4.2 Learning fairness data representation

Now that we have an identifiable and disentangled representation of the context c we will describe how we optimize the balanced fairness objective (BFO) in Eq. 1 efficiently when we only have access to biased unfair logged data. To this end, we design an adversarial learning strategy which comprises of two components: (1) A discriminator that is tasked to determine if the current representation of the disentangled context features satisfy "*Fair Exposure*" and (2) A representation learning function η which aims at learning balanced sensitive features that remove unfair factors in historical data. These new balanced sensitive representation $\eta(z_u, z_i)$ and non-sensitive features $\mathbf{z_n}$ are then used as inputs to a function f to predict the user feedbacks y. Note that f can be any recommendation model that uses feature embeddings as inputs.

To understand how we get to our final objective function, we will start by describing the discriminator in more detail. The proposed discriminator $D : \mathbb{R}^{l_u+l_i} \rightarrow \mathbb{R}^2$ is a classifier with a 2-dimensional softmax output layer, tasked to identify whether $\eta(\mathbf{z_u}, \mathbf{z_i})$ is exposed to the user, i.e. dimension 1 indicates the probability that the item was exposed to the user and dimension 0 indicates the probability that the item was not.

Intuitively, if the discriminator D is not able to determine whether item i was exposed to the user u based on $\eta(\mathbf{z_u}, \mathbf{z_i})$, we can conclude that the items were in fact exposed uniformly at random i.e. "*Fair Exposure*". In parallel to this classification task, we also train the representation function η and a recommendation model f to satisfy the user feedback prediction. This is to avoid trivial solutions learned through the classification task (adversarial process). Hence we arrive at the following objective ($\mathbf{c}$ is implicitly included in $\mathbf{z}$).

$$\begin{aligned} &L_b(\theta_{\boldsymbol{\phi}}, \theta_\eta, \theta_D, \theta_f) = L_b^f(\theta_{\boldsymbol{\phi}}, \theta_\eta, \theta_f) + L_b^D(\theta_{\boldsymbol{\phi}}, \theta_\eta, \theta_D) \\ &= \underbrace{\frac{1}{N}\sum_{t=1}^{N} o_i^t \delta(y^t, f(\eta(\mathbf{z_u^t}, \mathbf{z_i^t}), \mathbf{z_n^t}))}_{\text{User Feedback Prediction (BFO)}} + \underbrace{\gamma \sum_{i}^{m}\sum_{k=0}^{1} \mathbb{E}_{p_\eta(\mathbf{z_u}, \mathbf{z_i}|o_i=k)}[\log D^k(\eta(\mathbf{z_u}, \mathbf{z_i}))]}_{\text{Discriminator}} \end{aligned} \tag{8}$$

where D^k denotes the k-th element from the output of the discriminator. We optimize the above loss function in an alternating minmax game fashion as follows:

$$\min_{\theta_f, \theta_{\boldsymbol{\phi}}, \theta_\eta} \max_{\theta_D} L_b(\theta_{\boldsymbol{\phi}}, \theta_\eta, \theta_D, \theta_f), \; s.t. \sum_{k=0}^{1} D^k(\eta(\mathbf{z_u}, \mathbf{z_i})) = 1, \tag{9}$$

In order to understand why the representation $\eta(\mathbf{z_u}, \mathbf{z_i})$ is unbiased and how optimizing L_b achieves BFO based on $\eta(\mathbf{z_u}, \mathbf{z_i})$, we present the following theorem.

Theorem 4.1. *The minmax game on discriminator is defined by,*

$$\begin{aligned} &\min_{\theta_f, \theta_{\boldsymbol{\phi}}, \theta_\eta} \max_{\theta_D} \sum_{k=0}^{1} \mathbb{E}_{p_\eta(\mathbf{z_u}, \mathbf{z_i}|o_i=k)}[\log D^k(\eta(\mathbf{z_u}, \mathbf{z_i})], \\ &s.t. \sum_{k=0}^{1} D^k(\eta(\mathbf{z_u}, \mathbf{z_i})) = 1, \end{aligned} \tag{10}$$

and has optimal solution when $p(\eta(\mathbf{z_u}, \mathbf{z_i})|o_i = 1) = p(\eta(\mathbf{z_u}, \mathbf{z_i})|o_i = 0)$

In other words, the above theorem states, that under the assumption that we are able to optimize the in minimax game L_b^D of the discriminator, we are in fact in the setting of "*Fair Exposure*" wrt to η, i.e. $p(\eta(\mathbf{z_u}, \mathbf{z_i})|o_i = 1) = p(\eta(\mathbf{z_u}, \mathbf{z_i})|o_i = 0)$. Note that only optimizing L_b^f in Eq.8 without the discriminator loss L_b^D corresponds to learning the feedback from the context in the standard recommendation setting using biased unfair training data. By adding the discriminator loss L_b^D we enforce learning of a representation $\eta(\mathbf{z_u}, \mathbf{z_i})$ such that the "*Fair Exposure*" condition is upheld and thus BFO is optimized. Once we defined these representations of the sensitive attributes $\eta(\mathbf{z_u}, \mathbf{z_i})$, we learn the function f_η^* below. Note, the function f_η does not include $\mathbf{c}$ as it is implicitly included in $\eta(\mathbf{z_u}, \mathbf{z_i})$ and $\mathbf{z_n}$.

$$f_\eta^* = \arg\min_{f_\eta} \mathbb{E}_{o_i \sim p^{\text{uni}}(o_i|\mathbf{z_u}, \mathbf{z_i})} \mathbb{E}_{\mathbf{z_u}, \mathbf{z_i} \sim p(\mathbf{z_u}, \mathbf{z_i})} o_i [\delta(y, f_\eta(\eta(\mathbf{z_u}, \mathbf{z_i}), \mathbf{z_n}))].$$

4.3 Overall optimization objective

Now that we have described the two main stages of our proposed training process, we will show how to train our model B-FAIR in an end-to-end manner. To this end, we proposed the following objective function which is summarized below as a minmax game.

$$\begin{aligned} &\min_{\theta_f, \theta_{\boldsymbol{\phi}}, \theta_{\mathbf{h}}, \theta_\eta} \max_{\theta_D} L_{final}(\theta_D, \theta_f, \theta_{\boldsymbol{\phi}}, \theta_{\mathbf{h}}, \theta_\eta), \\ &s.t. \sum_{k=0}^{1} D^k(\eta(\mathbf{z_u}, \mathbf{z_i})) = 1 \end{aligned} \tag{11}$$

$$L_{final}(\theta_D, \theta_f, \theta_\phi, \theta_{\mathbf{h}}, \theta_\eta) = \underbrace{L_{dis}(\theta_\phi, \theta_{\mathbf{h}})}_{\text{Disentangle (Sec.4.1)}} + \underbrace{L_b(\theta_D, \theta_f, \theta_{\boldsymbol{\phi}}, \theta_\eta)}_{\text{Balanced Fairness (Sect.4.2)}}$$

As is common in adversarial/minmax games, we will alternate between maximizing wrt to θ_D and minimizing wrt to $\theta_f, \theta_\eta, \theta_{\boldsymbol{\phi}}$ and $\theta_{\mathbf{h}}$. Recall that our method aims at rectifying the unfairness problem from the data perspective using representation learning. In particular, we aim to get *balanced fairness*. We could easily add additional fairness constraints or change the downstream policy model f to obtain stricter constraints on fair prediction. However, we leave this for future work as it is not the focus of this work.

In summary, in this section, we described how to optimize the BFO in a two-stage representation learning process. We firstly, in section 4.1, describe a VAE-based disentanglement model which allows us to extract the sensitive features $\mathbf{z_u}, \mathbf{z_i}$ as well as the non-sensitive features $\mathbf{z_n}$ from our context data $\mathbf{c}$. By using the architecture proposed in [16], we can guarantee the identifiability of the extracted representations. In section 4.2, we then describe how we can use an adversarial training scheme to emulate the training

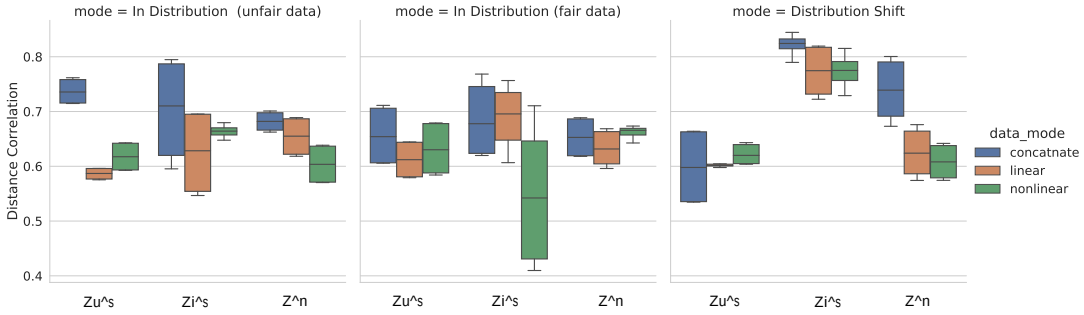


Figure 3: The distance correlation between the learned representation and the ground truth representation. If the distance correlation is well above 0.5 we have evidence that our latent representations are indeed identifiable.

Table 1: Summarization of the datasets.

Dataset	# User	# Item	Density	Domain
Synthetic	10000	32	1.756%	-
MovieLens	6,040	3,952	4.19%	Movie
Insurance	1,231	21	20.82%	Insurance

under the BFO objective, thus yielding balanced and fair representations which are not affected by the biased logged data. Lastly, we also prove theoretically that by optimizing our proposed objective, we achieve balanced fairness.

5 EXPERIMENTS

In this section, we conduct extensive experiments to demonstrate the effectiveness of our proposed method B-FAIR. In particular, we illustrate how B-FAIR outperforms current state-of-the-art debiasing methods on both synthetic as well as real-world data experiments. Furthermore, to get a better understanding, we also perform several ablation studies to investigate under which conditions our end-to-end algorithm can disentangle the context.

5.1 Experiment setup

5.1.1 ***Synthetic data:*** Similar to previous work [41, 47], we simulate 10,000 different users with 9 sensitive groups and 32 items with 3 sensitive groups, where each item has its own attributes. Each of the users and items consists of 32-dimension sensitive features and 32-dimension non-sensitive features, which are generated from a uniform distribution $U(-1, 1)$. Let $\mathbf{z_u^s}, \mathbf{z_i^s}$ describe the sensitive features of the user and item and $\mathbf{z_u^n}, \mathbf{z_i^n}$ be non-sensitive features of the user and item respectively. The context $\mathbf{c}_{ui}$ is generated as follows $\mathbf{c}_{ui} = l(\mathbf{z_u^s}, \mathbf{z_i^s}, \mathbf{z_u^n}, \mathbf{z_u^n})$, where we investigate several options for the function l: (1) A Concatenation of $\mathbf{z_u^s}, \mathbf{z_i^s}, \mathbf{z_u^n}$ and z_i^n, (2) a linear function (3) and non-linear function. The details of the exact functions are given later in section 5.4.

We report overall results using the concatenation function in Table 2 and further analyze the disentanglement of all three functions in section 5.4. After defining the context generation process, as in previous works [41, 47], we move on to how the scores are computed based on the context. In particular, as in [41, 47], for each item i, we define a score $r_{u,i}$ as follows, where the top-N exposure list is based on the score and we select 5 items from 32 item sets to the user.

$$r_{ui} = 1 - \sigma\left[\boldsymbol{a}^T \kappa_1\left(\kappa_2\left([\mathbf{c}_{ui}]\right)\right) + \mathcal{N}(0, 0.02)\right)\Big],$$

Based on the exposure list, we generate the feedback of a user u on an item i (in the exposure list) as follows:

$$\begin{aligned} x_1 &= \boldsymbol{a}^T \kappa_1\left(\kappa_2\left([\mathbf{c}_{ui}]\right)\right) + b, x_2 = \boldsymbol{a}^T \kappa_3\left(\kappa_2\left([-\mathbf{c}_{ui}]\right)\right) + b \\ y_{ui} &= \mathbb{I}\left(\sigma\left(x_1 + x_1 \cdot x_2\right)\right) \end{aligned} \tag{12}$$

$\kappa_1(\cdot)$ and $\kappa_2(\cdot)$ are piecewise functions, where $\kappa_1(x) = x - 0.5$ if $x > 0$, otherwise $\kappa_1(x) = 0.\kappa_2(x) = x$ if $x > 0$, otherwise $\kappa_2(x) = 0$. $\boldsymbol{a}^T$ is normalization term. Note that the above recommendation policy is unfair since the exposure list is influenced by both sensitive and non-sensitive context information i.e. $c = l(z_u^s, z_i^s, z_u^n, z_u^n)$. Hence this represents the setting where we are collecting biased unfair logged data on which we would then train our new recommendation policy. The resulting dataset will be our training/validation data. In order to evaluate how well B-FAIR performs we also generate a fair dataset for testing, which is generated using the below fair policy, as it does not use $\mathbf{z_u^s}, \mathbf{z_i^s}$ (the sensitive attributes):

$$r_{ui} = 1 - \sigma\left[\boldsymbol{a}^T \kappa_1\left(\kappa_2\left(\left[z_u^n, z_i^n\right]\right)\right) + \mathcal{N}(0, 0.02)\right)\Big] \tag{13}$$

In the above equation, the score function does not depend on the sensitive information $\mathbf{z_u^s}, \mathbf{z_i^s}$ and hence can be used as our fair ground truth to assess our method.

5.1.2 ***Real-world dataset***. **MovieLens-1M**[1]: The dataset contains user-item interactions and user profile information for movie recommendation. We use gender (binary classes) and movie tags (18 classes) as user and item-sensitive features respectively. **Insurance**[2]: The dataset is collected from insurance products recommendation system. We use the gender (binary classes) as user-sensitive information and item id (21 classes) as item-sensitive features. Here the sensitive information item id means that we consider individual fairness on the item side.

5.1.3 ***Metrics***. Since the prediction task is a binary classification, we consider the following standard evaluation metrics: AUC [27], ACC [14], Precision@N and Normalized Discounted Cumulative Gain (NDCG@N). For Movielens and synthetic datasets, we show Precision@5 and NDCG@5 and for the Insurance dataset, we show Precision@3 and NDCG@3. In the synthetic data, since we can

[1]https://grouplens.org/datasets/movielens/1m/
[2]https://www.kaggle.com/mrmorj/insurance-recommendation

Table 2: Comparisons on Synthetic and MovieLens. Our proposed method B-FAIR is outperforming most state-of-the-art baselines by the mean performance (± is the standard error of the results.).

Dataset	Strategy	MLP				GMF			
	Metrics	AUC ↑	ACC ↑	N@5 ↑	P@5 ↑	AUC ↑	ACC ↑	N@5 ↑	P@5 ↑
Synthetic	Base	.882±.003	.807±.002	.851±.003	.846±.001	.882±.003	.798±.002	.849±.001	.849±.001
	B-FAIR	**.916**±.003	**.827**±.003	.858±.002	**.863**±.001	**.907**±.002	**.822**±.001	**.861**±.003	**.858**±.002
	B-FAIR(-d)	.897±.003	.813±.003	.853±.003	.850±.001	.890±.004	.794±.001	.853±.001	.841±.002
	SNIPS	.902±.005	.815±.002	.851±.002	.843±.001	.894±.009	.813±.007	.857±.003	.853±.007
	IPS	.901±.004	.808±.007	.849±.001	.840±.002	.891±.006	.811±.006	.856±.001	.851±.003
	Direct	.813±.007	.768±.002	.834±.001	.806±.007	.708±.02	.581±.001	.811±.001	.768±.003
	DR	.908±.004	.813±.001	.853±.002	.841±.001	.890±.004	.803±.003	.858±.001	.855±.002
	CVIB	.913±.004	.817±.003	**.860**±.002	.859±.001	.889±.004	.807±.003	.857±.001	.853±.002
	ATT	.859±.020	.791±.003	.848±.002	.801±.005	.815±.058	.516±.006	.808±.001	.704±.003
	Metrics	AUC-F ↓	ACC-F ↓	N@5-F ↓	P@5-F ↓	AUC-F ↓	ACC-F ↓	N@5-F ↓	P@5-F ↓
MovieLens (gender)	Base	.0383±.001	.0234±.001	.0295±.001	.0046±.001	.0371±.001	.0244±.003	.0295±.001	.0041±.004
	B-FAIR	**.0317**±.001	**.0171**±.002	**.0227**±.001	**.0032**±.001	.0349±.001	.0165±.001	**.0208**±.004	**.0024**±.001
	SNIPS	.0370±.003	.0177±.004	.0268±.003	.0044±.001	.0395±.003	.0247±.003	.0297±.002	.0052±.001
	IPS	.0362±.006	.0227±.004	.0263±.004	.0033±.001	.0390±.003	.0246±.003	.0295±.002	.0040±.001
	Direct	.0373±.001	.0239±.002	.0263±.001	.0055±.002	**.0345**±.002	**.0155**±.002	.0243±.001	.0026±.002
	DR	.0358±.001	.0242±.001	.0287±.002	.0051±.001	.0361±.001	.0224±.001	.0252±.002	.0043±.001
	CVIB	.0369±.004	.0216±.002	.0275±.004	.0048±.001	.0415±.007	.0225±.003	.0319±.004	.0067±.001
	ATT	.0370±.001	.0182±.001	.0277±.002	**.0032**±.005	.0380±.003	.0194±.002	.0285±.003	.0072±.001
MovieLens (tags)	Base	.1421±.012	.2925±.024	.3943±.059	.4636±.010	.1277±.008	.2737±.023	.3698±.074	.4684±.002
	B-FAIR	.0981±.014	**.2175**±.021	**.2926**±.013	.4742±.006	**.0664**±.017	**.1700**±.032	**.3141**±.074	**.4205**±.007
	SNIPS	.1013±.041	.2325±.073	.3466±.072	.4679±.008	.0884±.019	.2037±.041	.3294±.060	.4631±.004
	IPS	.1360±.053	.2287±.071	.3210±.024	.4650±.008	.0897±.021	.1925±.043	.3541±.062	.4639±.003
	Direct	.0993±.031	.2687±.042	.3205±.031	**.4580**±.009	.1409±.058	.4543±.022	.3958±.032	.4658±.003
	DR	.0852±.023	.2275±.041	.3213±.022	.4699±.020	.0883±.013	.2425±.018	.3253±.062	.4641±.005
	CVIB	**.0943**±.054	.2350±.072	.3134±.031	.4637±.005	.0888±.011	.2350±.041	.3277±.083	.4679±.004
	ATT	.2055±.037	.2310±.032	.3552±.052	.4634±.007	.0997±.041	.4543±.031	.4512±.043	.4355±.011
Insuance	Base	0.0453±.003	0.0220±.002	0.0495±.003	0.0300±.002	0.0373±.005	0.0157±.001	0.0396±.004	0.0332±.003
	B-FAIR	**0.0186**±.002	**0.0191**±.002	**0.0180**±.002	0.0184±.003	**0.0233**±.003	0.0212±.002	**0.0219**±.003	**0.0204**±.003
	SNIPS	0.0347±.003	0.0246±.002	0.0343±.003	0.0172±.004	0.0279±.002	0.0246±.002	0.0243±.003	0.0273±.005
	IPS	0.0202±.003	0.0246±.001	0.0227±.003	0.0201±.005	0.0513±.002	**0.0157**±.001	0.0435±.003	0.0210±.005
	Direct	0.0650±.004	0.0216±.002	0.0542±.004	0.0213±.004	0.0394±.004	0.0157±.001	0.0237±.004	0.0330±.003
	DR	0.0231±.002	0.0246±.002	0.0197±.002	**0.0159**±.001	0.0316±.002	0.0243±.002	0.0232±.002	0.0273±.003
	CVIB	0.0188±.002	0.0323±.002	0.0200±.003	0.0208±.001	0.0465±.003	0.0246±.003	0.0427±.004	0.0402±.003
	ATT	0.0221±.003	0.0206±.002	0.0273±.003	0.0240±.003	0.0258±.001	0.0183±.004	0.0275±.004	0.0238±.003

generate the fair test dataset, the scores given in table 2 reflect whether we were able to learn a unbiased recommendation system.

In the real-world dataset, similar to [18], we use metrics ACC-F, AUC-F, NDCG@N-F and Precision@N-F, where we evaluate the scores (ACC, AUC, Precision@N, NDCG@N) for each sensitive group and calculate the discrepancy between the highest and lowest score (e.g. AUC in male group and female group). By achieving fairness in the training process we are inherently also reducing the performance gap (e.g. ACC-F, AUC-F, NDCG@N-F and Precision@N-F).

5.1.4 ***Baselines***. As mentioned in the related works (Section 2), there are two parts to the fairness pipeline proposed by [11]: (1) Removing the data bias and (2) then improving fairness in the online system through constraints. Since the objective of our framework is to achieve data debiasing, we mainly focus on the debiasing literature for our baseline experiments.

These baselines include the "*Base model*" (the model without any debiasing) and the state-of-the-art debiasing methods such as Inverse Propensity Score (IPS) [32], Self-nomarlized IPS (SNIPS) [33], Doubly Robust (DR) [10], ATT [28] and CVIB [35]. The baselines IPS and SNIPS are IPS-based methods, where the pretrained propensity weight evaluation model is required. Direct and ATT are both Direct-learning-based methods, where an imputation model to generate a counterfactual data samples is required. DR method is the combination of IPS and Direct method and CVIB is a representation learning based method. By reformulating the BFO into a debiasing problem we can use a similar objective to ours. The only difference is that they aim to achieve uniform exposure in all contexts, whereas we aim to achieve fair exposure conditioned on the sensitive group. To achieve a fair comparison, we add the user and item-sensitive group index as a part of the context feature to each baseline. For IPS based method, we use the fairness representation learned from the disentanglement method to calculate the propensity score.

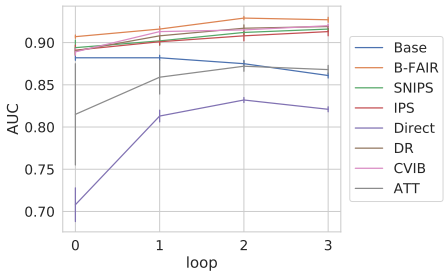


Figure 4: Performance over 3 feedback loops

5.2 Implementation Details

For MovieLens and Insurance dataset, we split the original dataset by 80% sample of train 10% of validation and 10% of test. For the simulation dataset, the ratio between the training and testing (including validation) sets is controlled as 3:1. We re-sample the data to make sure every sensitive group has same number of samples in test set and show statistical information in Table 1. The weighting parameter λ and γ are determined in range of $[.001, 0.5]$. The user/item embedding dimension is empirically set as 32.

We apply our framework to different base models including **MLP** and **GMF** [15]. All the experiments are conducted based on a server with a 16-core CPU, 128g memory and RTX 5000 GPU. We specify different parts of objective (11) as follows: δ is implemented by the binary cross-entropy loss to model user feedback. The categorical features for the users are encoded by embedding matrix. The continuous features are directly multiplied by weighting matrices to derive the representation. Denoting V_k, A_k and B_k as weighting parameters, σ as softmax function with 2 dimension output and ELU are activation functions. The disentanglement function $\boldsymbol{\phi}$ consist of three separate functions $\boldsymbol{\phi}_u(\mathbf{c}), \boldsymbol{\phi}_i(\mathbf{c}), \boldsymbol{\phi}_n(\mathbf{c})$ to infer $\mathbf{z_u}, \mathbf{z_i}, \mathbf{z_n}$, respectively. Each of neural network is designed as $V_1\text{ELU}(V_2\text{ELU}(V_3[\mathbf{c}]))$. The architecture of generate function is $A_1\text{ELU}(A_2\text{ELU}(A_3[\mathbf{z_i}, \mathbf{z_u}, \mathbf{z_n}]))$. The balanced representation function is implemented using a linear layer. The discriminitor function is defined as $D(\phi(u_t, z_t)) = \sigma(B_1\text{ELU}(B_2\eta(z_u, z_i)))$.

5.3 Overall comparison

In this section, we present our results in table 2. In most cases, the performance of the direct-based method (Direct, ATT) is worse than that of IPS or SNIPS. This is because the imputation model is biased by the dataset since the dataset is disturbed by a complex environment and thus the performance of the imputation model cannot be guaranteed. CVIB and DR can generally achieve better performance than IPS, SNIPS and Direct and ATT, which is consistent with the previous work [10]. In most cases, the state-of-art representation learning-based method CVIB perform better than other baselines, because the model considers a more general setting of the real system to avoid noise interference.

Synthetic Data Results: Taking a closer look at each of the experiments separately: Firstly, in the synthetic data, our proposed method B-FAIR obtained significant performance improvements compared to the base model across all 4 metrics AUC, ACC, NDCG@5 and Precision@5 (see table 2). Note that, we generated fair and unbiased synthetic test data according to Eq. 13. Thus the performance on standard metrics allows us to determine whether we achieve less biased prediction. To further investigate each component of our two-stage method, we also performed experiments where we did not apply a disentanglement step (B-FAIR(-d)). As can be seen in Table 2, B-FAIR(-d) does not perform on par with B-FAIR and hence we conclude that the disentanglement step is crucial to the success of B-FAIR. These experiments in controlled environments, validate our hypothesis that our method B-FAIR allows us to train a recommendation policy as if the logged data was balanced fair.

Finally, we examined the scenario in which the recommendations at time $t+1$ are contingent on the recommendations at time t, i.e. sequential recommendation with feedback loops. To simulate this scenario, we created a synthetic data example with a feedback loop. We updated the recommendation list using the current policy after 50 epochs of training. The exposure list at time $t+1$ was constructed based on the score r_{ui}, which was calculated from the previous recommendation model at time t. The recommendation list was then re-labeled using the rule outlined in Eq. 12. The results in Figure 4 demonstrate that our method B-FAIR outperforms other methods over multiple feedback loops. In particular, the B-FAIR method achieved the best fairness performance, with an AUC of 0.927 after three loops, while the base method failed to maintain fairness as the performance declined with each loop.

Real-World Data Results: For the real-world dataset MovieLens and Insurance, we demonstrate the performance on the fairness scores AUC-F, ACC-F, NDCG@5-F and Precision@5-F on user sensitive group gender and item sensitive group movie tags for MovieLens and gender sensitive group for Insurance. Given that we do not have access to a balanced dataset as in our synthetic data experiments, we use these metrics as proxies to illustrate B-FAIR. This means the lower the fairness scores, the smaller the discrepancy between the sensitive group and therefore the fairer the method.

Compared with base method, we get a fairness performance improvement of {AUC-F: 11.6% ACC-F: 29.6% NDCG@5-F: 26.3% Precision@5-F: 26.3% } on user sensitive group (gender), {AUC-F: 39.4% ACC-F: 31.7% NDCG@5-F: 20.4% Precision@5-F: 3.9% } on item sensitive group (movie tags) on MovieLens dataset and The performance on general scores is close to the current optimal. Although the general scores can not be applied to measure fairness performance, the results show that our method can enhance fairness performance and not damage user feedback prediction.

5.4 Ablation study

Ablation Study Setup: For our ablation study on the identifiability of our disentanglement stage, we consider the distance correlation between the features learned by disentanglement method $\hat{\mathbf{z}}$ and $\mathbf{z}$, which is defined as $d\,\text{corr}(\hat{\mathbf{z}}, \mathbf{z}) = \frac{\hat{d}\,\text{cov}(\mathbf{z},\mathbf{z})}{\sqrt{d\,\text{cov}(\mathbf{z},\hat{\mathbf{z}})\hat{d}\,\text{cov}(\hat{\mathbf{z}},\hat{\mathbf{z}})}}$. $\hat{d}\,\text{cov}$ in $d\,\text{corr}$ describes the covariance between two variables $(\mathbf{z}, \hat{\mathbf{z}})$. The score measures the distance between two distributions, where higher distance correlation means better disentanglement of the method.

In other words, if the distance correlation between the learned sensitive features $\hat{\mathbf{z}}^{\mathbf{s}}_{\mathbf{u}}, \hat{\mathbf{z}}^{\mathbf{s}}_{\mathbf{i}}$ and the ground truth sensitive features $\mathbf{z}^{\mathbf{s}}_{\mathbf{u}}, \mathbf{z}^{\mathbf{s}}_{\mathbf{i}}$ is high, it means that we can capture most of the sensitive information in our learned representation. We equivalently also show the score between the learned non-sensitive feature $[\hat{\mathbf{z}}^{\mathbf{n}}_{\mathbf{u}}, \hat{\mathbf{z}}^{\mathbf{n}}_{\mathbf{i}}]$ and ground truth $[\mathbf{z}^{\mathbf{n}}_{\mathbf{u}}, \mathbf{z}^{\mathbf{n}}_{\mathbf{i}}]$.We study the quality of the disentanglement stage under different context generative functions. We base

this experiment on synthetic data since it allows us to flexibly set the context $\mathbf{c}$ generation function from hidden features $\mathbf{z}$.

Specifically, we design three functions, (1) A concatenation function $\mathbf{c} = [\mathbf{z^n}, \mathbf{z_i^s}, \mathbf{z_u^s}]$, (2) A linear function $\mathbf{c} = \mathbf{A}(\mathbf{z^n}, \mathbf{z_i^s}, \mathbf{z_u^s}) + b$, where $\mathbf{A}$ is a randomly generated invertible matrix and b is random vector sampled from uniform distribution $U(-1, 1)$ and (3) A nonlinear generative process where we add an additional sigmoid layer based on linear function. We report the results on distance correlation score on user-sensitive feature $\mathbf{z_u^s}$, item-sensitive feature $\mathbf{z_i^s}$ and non-sensitive $\mathbf{z_n}$ in fig. 3.

Ablation Study Results: Note that the higher the distance correlation, the more information our disentangling step was able to extract. In addition, to validate that this stage works even under distribution shift, we added three different plots. In Figure 3, "*In distribution*" (fair data) means we use fair data for training and testing and "*Distribution Shift*" denotes that we are using unfair data to train and fair data to test. In general, the bars in fig. 3 are well above 0.5 indicating that we are learning disentangled representation and that they are heavily correlated with the ground truth latent variables. We also note that when $\mathbf{c} = [\mathbf{z^n}, \mathbf{z_i^s}, \mathbf{z_u^s}]$, i.e. concatenation, the disentangling seems to work the best i.e. highest distance correlations, as it is also the easiest. This is followed by the linear and the non-linear mixture functions. These results illustrate that we can obtain disentangled representations.

6 THEORETICAL PROOF

6.1 Proof of Theorem 4.1

Suppose $\mathbf{s} = \eta(\mathbf{z_u}, \mathbf{z_i})$ and $p_\eta^0(\mathbf{s}) = p(\eta(\mathbf{z_u}, \mathbf{z_i}|o_i = 0)$ and $p_\eta^1(\mathbf{s}) = p(\eta(\mathbf{z_u}, \mathbf{z_i}|o_i = 1)$ then objective function in Theorem 4.1 can be written as:

$$\min_{\eta} \max_{\theta_D} \int_{\mathbf{s}} \sum_{k=0}^{1} p_\eta^i(\mathbf{s}) \log D^k(\mathbf{s}) d\mathbf{s}, \quad \text{s.t.} \quad \sum_{k=0}^{1} D^k(\eta(\mathbf{z_u}, \mathbf{z_i})) = 1 \tag{14}$$

We apply Lagrange multiplier to solve the above constraint optimization problem:

$$L(D, \lambda) = \min_{\eta} \max_{\theta_D} \int_{\mathbf{s}} \sum_{k=0}^{1} p_\eta^k(\mathbf{s}) \log D(\mathbf{s}) d\mathbf{s} + \lambda \left(\sum_{k=0}^{1} D^k(\mathbf{s}) - 1 \right)$$

The solution of above are $D^k(\mathbf{s}) = -\frac{p_\eta^k(\mathbf{s})}{\lambda}$ and $\lambda = -\sum_{k=0}^{1} p_\eta^k(\mathbf{s})$ when $\frac{\partial L(D,\lambda)}{\partial D} = 0$, $\frac{\partial L(D,\lambda)}{\partial \lambda} = 0$. Apply optimal $D^k(\mathbf{s})$ into Eq. 14, we get the following derivation:

$$\begin{aligned} \text{Eq. 14} &= \sum_{k=0}^{1} \int_{\mathbf{s}} p_\eta^k(\mathbf{s}) \log \frac{p_\eta^k(\mathbf{s})}{\sum_{k=0}^{1} p_\eta^k(\mathbf{s})} d\mathbf{s} \\ &= \sum_{k=0}^{1} D_{KL} \left(p_\eta^k(\mathbf{s}) \| \frac{1}{2} \sum_{k=0}^{1} p_\eta^k(\mathbf{s}) \right) + \log 2 = 2\text{JSD} \left(p_\eta^1(\mathbf{s}), p_\eta^2(\mathbf{s}) \right) \end{aligned}$$

Where $\text{JSD}\left(p_\eta^1(\mathbf{s}), p_\eta^2(\mathbf{s})\right)$ is multivariate Jensen-Shannon Divergence. The above equation is minimized when $p_\eta^1(\mathbf{s}) = p_\eta^2(\mathbf{s})$ thus we get the theoretical result.

6.2 Proof of Eq. 6

We decompose the KL term in ELBO in Eq. 7 by Eq. 3 as:

$$\begin{aligned} &\mathcal{D}_{KL}(q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i) \| p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z}|g_u, g_i))] \\ &= \iiint q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i) \log \frac{q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i)}{p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z_u}|g_u)} \\ &\quad + q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i) \log \frac{q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i)}{p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z_u}|g_u)} \\ &\quad + q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i) \log \frac{q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i)}{p(\mathbf{z_n})} - 2H(\mu_{\theta_\phi}(\mathbf{c}, g_u, g_i), \sigma I) \end{aligned} \tag{15}$$

Since $\boldsymbol{\phi}$ is injective function which infer (non-)sensitive features separately by using three independent conditional prior and we add an additional term L_n to guarantee the independence between $\mathbf{z_n}$ and $\mathbf{z_u}$, $\mathbf{z_u}$, we suppose the following function hold.

$$q_{\boldsymbol{\theta_\phi}}(\mathbf{z} \mid \mathbf{c}, g_u, g_i) = q_{\boldsymbol{\theta_\phi}}(\mathbf{z_u} \mid \mathbf{c}, g_u, g_i) q_{\boldsymbol{\theta_\phi}}(\mathbf{z_i} \mid \mathbf{c}, g_u, g_i) q_{\boldsymbol{\theta_\phi}}(\mathbf{z_n} \mid \mathbf{c}, g_u, g_i)$$

Then for each terms in Eq.15 by above decomposition we have:

$$\begin{aligned} &\iiint q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i) \log \frac{q_{\boldsymbol{\theta_\phi}}(\mathbf{z}|\mathbf{c}, g_u, g_i)}{p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z_u}|g_u)} d\mathbf{z_u} d\mathbf{z_i} d\mathbf{z_n} \\ &= \int q_{\boldsymbol{\theta_\phi}}(\mathbf{z_u}|\mathbf{c}, g_u, g_i) \log \frac{q_{\boldsymbol{\theta_\phi}}(\mathbf{z_u}|\mathbf{c}, g_u, g_i)}{p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z_u}|g_u)} \iint q_{\boldsymbol{\theta_\phi}}(\mathbf{z_i}, \mathbf{z_n}|\mathbf{c}, g_u, g_i) d\mathbf{z_u} d\mathbf{z_i} d\mathbf{z_n} \\ &+ \int q_{\boldsymbol{\theta_\phi}}(\mathbf{z_u}|\mathbf{c}, g_u, g_i) \iint q_{\boldsymbol{\theta_\phi}}(\mathbf{z_i}, \mathbf{z_n}|\mathbf{c}, g_u, g_i) \log q_{\boldsymbol{\theta_\phi}}(\mathbf{z_i}, \mathbf{z_n}|\mathbf{c}, g_u, g_i) d\mathbf{z_u} d\mathbf{z_i} d\mathbf{z_n} \\ &= \mathcal{D}_{kl} \left(q_{\boldsymbol{\theta_\phi}}(\mathbf{z_u} \mid \mathbf{c}, g_u, g_i) \| p_{\boldsymbol{\lambda_1}, \boldsymbol{\lambda_2}}(\mathbf{z_u} \mid g_u) \right) \end{aligned}$$

Similarly for other terms in Eq.15, the results lead to Eq. 6.

7 CONCLUSION AND LIMITATIONS

In this paper, we tackle the problem of unfair feedback loops due to biased and unfair historical data. Given that recommendation models are often trained on the data that they themselves produce, unfair patterns in the data can easily be amplified. Hence in this work, we firstly proposed a new fairness objective coined "*Balanced Fairness Objective*" (BFO) and secondly present a two-stage end-to-end algorithm to learn the balanced fair representation. Under this new definition of fairness, we aim to train a recommendation model, *as if the training data came from a uniform recommendation policy conditioned on sensitive attributes*. We show in extensive synthetic as well as real-world experiments that our proposed method B-FAIR achieves state-of-the-art performance compared to other methods.

We believe that this paper opens a new door for representation learning in fair/debiased recommendation settings. However, there are still limitations, which we aim to improve in future work. (1) We give a theoretical analysis of how we can optimise BFO using a minmax objective. However, in real-world experiments, sample size and data noise will influence the performance. Hence, we plan to extend our theorem and algorithms with sample complexities and robustness guarantees. (2) Since B-FAIR focuses on rectifying the unfair and biased data problem using representation learning, future work would look into combining both stages in the fairness pipeline mentioned in [11]. This means, that we could potentially add fairness constraints to our BFO objective. We leave this exciting direction for future work as it is out of the scope of this paper.

REFERENCES

[1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. 2018. A reductions approach to fair classification. In *International Conference on Machine Learning*. PMLR, 60–69.

[2] Qingyao Ai, Keping Bi, Cheng Luo, Jiafeng Guo, and W Bruce Croft. 2018. Unbiased learning to rank with unbiased propensity estimation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 385–394.

[3] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H Chi, et al. 2019. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2212–2220.

[4] Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. AutoDebias: Learning to Debias for Recommendation. *arXiv preprint arXiv:2105.04170* (2021).

[5] 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* abs/2010.03240 (2020). arXiv:2010.03240

[6] Andrew Cotter, Heinrich Jiang, and Karthik Sridharan. 2019. Two-player games for efficient non-convex constrained optimization. In *Algorithmic Learning Theory*. PMLR, 300–332.

[7] Jingtao Ding, Yuhan Quan, Xiangnan He, Yong Li, and Depeng Jin. 2019. Reinforced Negative Sampling for Recommendation with Exposure Data.. In *IJCAI*. 2230–2236.

[8] Sihao Ding, Fuli Feng, Xiangnan He, Yong Liao, Jun Shi, and Yongdong Zhang. 2021. Causal incremental graph convolution for recommender system retraining. *arXiv preprint arXiv:2108.06889* (2021).

[9] Zhenhua Dong, Hong Zhu, Pengxiang Cheng, Xinhua Feng, Guohao Cai, Xiuqiang He, Jun Xu, and Jirong Wen. 2020. Counterfactual learning for recommender system. In *Fourteenth ACM Conference on Recommender Systems*. 568–569.

[10] Miroslav Dudík, John Langford, and Lihong Li. 2011. Doubly Robust Policy Evaluation and Learning. In *ICML 2011*. Omnipress, 1097–1104.

[11] Michael D Ekstrand, Robin Burke, and Fernando Diaz. 2019. Fairness and discrimination in recommendation and retrieval. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 576–577.

[12] Ruoyuan Gao and Chirag Shah. 2020. Counteracting bias and increasing fairness in search and recommender systems. In *Fourteenth ACM Conference on Recommender Systems*. 745–747.

[13] Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. 2016. Satisfying real-world goals with dataset constraints. *NeurIPS* 29 (2016).

[14] Asela Gunawardana and Guy Shani. 2009. A survey of accuracy evaluation metrics of recommendation tasks. *JMLR* 10, 12 (2009).

[15] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[16] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. 2020. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2207–2217.

[17] Karl Krauth, Yixin Wang, and Michael I Jordan. 2022. Breaking Feedback Loops in Recommender Systems with Causal Inference. *arXiv preprint arXiv:2207.01616* (2022).

[18] Caitlin Kuhlman, MaryAnn VanValkenburg, and Elke Rundensteiner. 2019. Fare: Diagnostics for fair ranking using pairwise error metrics. In *The world wide web conference*. 2936–2942.

[19] Carolin Lawrence, Artem Sokolov, and Stefan Riezler. 2017. Counterfactual Learning from Bandit Feedback under Deterministic Logging:A Case Study in Statistical Machine Translation. In *EMNLP*.

[20] Jurek Leonhardt, Avishek Anand, and Megha Khosla. 2018. User fairness in recommender systems. In *Companion Proceedings of the The Web Conference 2018*. 101–102.

[21] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2021. Personalized Counterfactual Fairness in Recommendation. *arXiv preprint arXiv:2105.09829* (2021).

[22] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2021. Towards personalized fairness based on causal notion. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1054–1063.

[23] Yunqi Li, Yingqiang Ge, and Yongfeng Zhang. 2021. Tutorial on fairness of machine learning in recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2654–2657.

[24] Benjamin M. Marlin and Richard S. Zemel. 2009. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, USA, October 23-25, 2009*. ACM, 5–12.

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

[26] Harikrishna Narasimhan. 2018. Learning with complex loss functions and constraints. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1646–1654.

[27] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[28] Yuta Saito. 2020. Asymmetric Tri-training for Debiasing Missing-Not-At-Random Explicit Feedback. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*. ACM, 309–318.

[29] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.

[30] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. In *ICML (JMLR Workshop and Conference Proceedings, Vol. 48)*. JMLR.org, 1670–1679.

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

[32] Adith Swaminathan and Thorsten Joachims. 2015. Counterfactual Risk Minimization: Learning from Logged Bandit Feedback. *CoRR* abs/1502.02362 (2015). arXiv:1502.02362

[33] Adith Swaminathan and Thorsten Joachims. 2015. The Self-Normalized Estimator for Counterfactual Learning. In *NIPS*. 3231–3239.

[34] Yu Wang, Xin Xin, Zaiqiao Meng, Xiangnan He, Joemon Jose, and Fuli Feng. 2021. Probabilistic and Variational Recommendation Denoising. *arXiv preprint arXiv:2105.09605* (2021).

[35] Zifeng Wang, Xi Chen, Rui Wen, Shao-Lun Huang, Ercan E. Kuruoglu, and Yefeng Zheng. 2020. Information Theoretic Counterfactual Learning from Missing-Not-At-Random Feedback. In *NeurIPS 2020*.

[36] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1791–1800.

[37] Blake Woodworth, Suriya Gunasekar, Mesrob I Ohannessian, and Nathan Srebro. 2017. Learning non-discriminatory predictors. In *Conference on Learning Theory*. PMLR, 1920–1953.

[38] Lin Xiao, Zhang Min, Zhang Yongfeng, Gu Zhaoquan, Liu Yiqun, and Ma Shaoping. 2017. Fairness-aware group recommendation with pareto-efficiency. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*. 107–115.

[39] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 279–287.

[40] Mengyue Yang, Guohao Cai, Furui Liu, Zhenhua Dong, Xiuqiang He, Jianye Hao, Jun Wang, and Xu Chen. 2022. Debiased Recommendation with User Feature Balancing. *arXiv preprint arXiv:2201.06056* (2022).

[41] Mengyue Yang, Quanyu Dai, Zhenhua Dong, Xu Chen, Xiuqiang He, and Jun Wang. 2021. Top-N Recommendation with Counterfactual User Preference Simulation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2342–2351.

[42] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. 2021. Causalvae: Disentangled representation learning via neural structural causal models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9593–9602.

[43] Sirui Yao and Bert Huang. 2017. Beyond parity: Fairness objectives for collaborative filtering. *NeurIPS* 30 (2017).

[44] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *International conference on machine learning*. PMLR, 325–333.

[45] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal Intervention for Leveraging Popularity Bias in Recommendation. *arXiv preprint arXiv:2105.06067* (2021).

[46] Zihao Zhao, Jiawei Chen, Sheng Zhou, Xiangnan He, Xuezhi Cao, Fuzheng Zhang, and Wei Wu. 2021. Popularity Bias Is Not Always Evil: Disentangling Benign and Harmful Bias for Recommendation. *arXiv preprint arXiv:2109.07946* (2021).

[47] Hao Zou, Kun Kuang, Boqi Chen, Peixuan Chen, and Peng Cui. 2019. Focused context balancing for robust offline policy evaluation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 696–704.