Average User-side Counterfactual Fairness for Collaborative Filtering

PENGYANG SHAO, Hefei University of Technology, China
LE WU*, Hefei University of Technology, China
KUN ZHANG, Hefei University of Technology, China
DEFU LIAN, University of Science and Technology of China, China
RICHANG HONG, Hefei University of Technology, China
YONG LI, Tsinghua University, China
MENG WANG*, Hefei University of Technology, China

Recently, the user-side fairness issue in Collaborative Filtering (CF) algorithms has gained considerable attention, arguing that results should not discriminate an individual or a sub user group based on users' sensitive attributes (e.g., gender). Researchers have proposed fairness-aware CF models by decreasing statistical associations between predictions and sensitive attributes. A more natural idea is to achieve model fairness from a causal perspective. The remaining challenge is that we have no access to interventions, i.e., the counterfactual world that produces recommendations when each user have changed the sensitive attribute value. To this end, we first borrow the Rubin-Neyman potential outcome framework to define average causal effects of sensitive attributes. Then, we show that removing causal effects of sensitive attributes is equal to average counterfactual fairness in CF. Then, we use the propensity re-weighting paradigm to estimate the average causal effects of sensitive attributes and formulate the estimated causal effects as an additional regularization term. To the best of our knowledge, we are one of the first few attempts to achieve counterfactual fairness from the causal effect estimation perspective in CF, which frees us from building sophisticated causal graph. Finally, experiments on three real-world datasets show the superiority of our proposed model.

CCS Concepts: • **Information systems** → *Collaborative filtering*; **Recommender systems**.

Additional Key Words and Phrases: collaborative filtering, potential outcome framework

ACM Reference Format:
Pengyang Shao, Le Wu, Kun Zhang, Defu Lian, Richang Hong, Yong Li, and Meng Wang. 2018. Average User-side Counterfactual Fairness for Collaborative Filtering. *J. ACM* 37, 4, Article 111 (August 2018), 26 pages. https://doi.org/XXXXXXX.XXXXXXX

*Corresponding authors.

Authors' addresses: Pengyang Shao, shaopymark@gmail.com, Hefei University of Technology, No. 485 Danxia Road, Hefei, China, 230601; Le Wu, lewu.ustc@gmail.com, Hefei University of Technology, No. 485 Danxia Road, Hefei, China, 230601; Kun Zhang, zhkun@hfut.edu.cn, Hefei University of Technology, No. 485 Danxia Road, Hefei, China, 230601; Defu Lian, liandefu@ustc.edu.cn, University of Science and Technology of China, No. 443 Huangshan Road, Hefei, China; Richang Hong, hongrc.hfut@gmail.com, Hefei University of Technology, No. 485 Danxia Road, Hefei, China, 230601; Yong Li, liyong07@tsinghua.edu.cn, Tsinghua University, No. 30 Shuangqing Road, Beijing, China; Meng Wang, eric.mengwang@gmail.com, Hefei University of Technology, No. 485 Danxia Road, Hefei, 230601, China.

ACM 0004-5411/2018/8-ART111
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Recommender systems are gaining critical impacts on humans and society, shaping the movies we watch, the news we read, the job we seek, etc [31, 58, 59]. As one of the representative approaches for recommender systems, Collaborative Filtering (CF) has been widely deployed in many scenarios due to relatively high performance and easy-to-collect user behavior data [70].

The fairness issues in CF algorithms are widespread and urgently need to be addressed[27, 69, 75]. For example, the OECD's Programme for International Student Assessment (PISA) collects student behaviors from the real world, however, the collected data display obvious gender discrimination in student behaviors [37]. Note that, CF algorithms have been widely used to obtain students' ability representations from their interaction behaviors with exercises [36, 60, 74]. In this way, CF models trained on such data would be unfair [9, 10, 72], leading to further education inequalities in the real world, e.g., exercise recommendation, class assignment, and even admission.

Among all fairness definitions, group fairness has been widely used to measure the treatment differences between the under-represented group and the over-represented group. Generally, group fairness requires that CF algorithms should not discriminate or favor an individual or a sub group based on users' sensitive attributes (e.g., gender and race) [47]. To achieve this goal, researchers have designed various group fairness principles from a statistical perspective, such as demographic parity and equality of opportunity [21, 30], and have proposed user fairness-oriented CF models [5, 75]. For example, fairness-aware regularization terms are proposed to decouple complicated correlations between sensitive attributes and recommendation results [75]. Adversarial training has been widely adopted for user fairness issues as it can ensure that sensitive attributes are orthogonal to user embeddings [5, 69, 79]. Nowadays, researchers have proposed FairGNN, which simultaneously employs a graph based sensitive attribute estimator and an adversarial learning based module to improve fairness performance of graph based models [13]. In summary, data-driven models achieve fairness by decreasing statistical correlations between sensitive attributes and predicted results.

However, some researchers argue that statistical fairness metrics have the potential to actually increase discrimination [12, 29, 71]. As Randomized Controlled Trial (RCT) is considered as the gold standard in scientific experiments, a more natural idea is to measure the effects of sensitive attributes in predictions and model fairness from a causal perspective. By using a causal vocabulary, researchers have designed *counterfactual fairness* as follows: an algorithm (e.g., a CF algorithm) is counterfactually fair if the prediction results (e.g., CF outputs) are the same between the factual world and an imagined counterfactual world where users' sensitive attributes had changed [29]. The challenge lies in that we only have access to observational data in the factual world where each user only belongs to one sensitive attribute category. To address the challenge, researchers leveraged the causal framework of Pearl [46] to model relationships between user sensitive attributes and the prediction results, and utilized a latent exogenous variable to represent user characteristics unrelated to sensitive attributes. Given a pre-defined causal graph stating the causal relationships among variables, researchers have successfully achieved causal fairness [12, 17, 33, 35, 62]. For example, Chiappa et al. proposed a path-specific counterfactual fairness to distinguish the effects of different paths in a causal graph and eliminate causal effects along specific unfair paths for fairness [12]. These existing causal approaches all assume that the causal graph is available. However, we argue that Pearl's causal framework cannot be applied to user-side fairness issues in CF models. Note that, high-dimension interaction data is the only and necessary input for CF models. It is a challenging task even for a domain expert to find a proper latent exogenous high-dimension variable to represent interactive information independent of sensitive attributes. This problem becomes more severe as most users do not expose their sensitive attributes in recommender systems, as explicit sensitive attribute values are necessary for the modeling of latent exogenous variables [29].

To this end, instead of leveraging Pearl's causal graphs, we start from Rubin-Neyman potential outcome framework to estimate causal effects of sensitive attributes based on users' behavior data [25, 53, 56]. After that, we prove that average counterfactual user fairness can be achieved by minimizing the causal effects of user sensitive attributes on predicted results. Thus, our target turns to minimize the estimated causal effects. As most users are unwilling to expose their sensitive attributes, we further propose how to exploit the user-item interaction bipartite graph structure, and design a graph self-supervised learning model to better estimate propensities and predict missing user sensitive attributes with limited sensitive information. Then, we use well-estimated propensities as weights for data samples to estimate the causal effects of sensitive attributes on predicted results. After that, we formulate the estimated causal effects as an additional regularization term in the predicted optimization process. In this way, we can optimize the CF model and simultaneously achieve average counterfactual user fairness. To the best of our knowledge, we are one of the first few attempts to achieve counterfactual fairness from the causal effect estimation perspective, which frees us from the usual causal fairness approaches that rely on building sophisticated causal graphs. Finally, we conduct extensive experiments on two real-world datasets to demonstrate the effectiveness of our proposed a *C*ounter*F*actually *Fair* collaborative filtering model (*CFFair*).

In summary, the key contributions of this paper are listed as follows:

- To the best of our knowledge, we are one of the first few attempts to achieve average user-side counterfactual fairness from the causal effect estimation perspective for CF models.
- We develop a novel graph self-supervised propensity estimator for reweighing data samples, which is in favor of estimating average causal effects with limited sensitive information.
- We conduct extensive experiments on three real-world datasets. Experimental results clearly show the effectiveness of our proposed *CFFair* on achieving fairness.

2 RELATED WORKS

2.1 CF based Recommender Systems

Recommender systems have been widely used to help users find potential items of interest [44]. Among recommender systems, CF models have been widely adopted in recommender systems due to their relatively high performance and easy-to-collect high-dimension interactive data. The interactive data denotes user behaviors on different items, e.g., clicks and purchase (implicit feedback) [50], and ratings (explicit feedback) [44]. Learning high-quality embeddings from interactive data is the key to successful CF models [23, 32, 73]. The predicted behavior can be represented as the inner dot [50] or a neural network's output [64] of corresponding user embedding and item embedding. Typically, there are two types of embedding learning based methods. The classical latent factor based models utilize matrix factorization methods to learn free user and item embeddings. Most users implicitly express their item preferences, e.g., click or purchase. Bayesian Personalized Ranking (BPR) focuses on the ranking issues of unobserved items in implicit feedback [50]. Behavior data in recommender systems naturally form a user-item bipartite graph. Therefore, graph-based models for CF have been popular in the CF research community [23, 32]. This process relies on iteratively updating user (item) embedding from their neighborhood's item (user) embeddings [26].

Empirically, these graph-based models perform better than classical latent factor based models because of utilizing the high-order collaborative signals in the user-item bipartite graph [7, 16, 39, 64]. Researchers notice that classic GCN based models could not model deeper layers due to the over smoothing effects, which causes decrease of recommendation performance. LR-GCCF has been proposed to enhance recommendation performance by removing non-linearities and proposing a residual network structure [32]. These two operations can successfully alleviate the over smoothing problem in graph convolution aggregation operation with sparse user-item interaction

data. In addition, self-supervised learning has been proven effective in produce high-quality graph representations of good generalizability, transferability, and robustness even without designing sophisticated GNN architectures [68, 77].

2.2 Statistical Approaches for User-side Fairness in CF

As artificial intelligence applications have been applied in modern society, researchers show great interest in whether applications are in compliance with legal and ethical requirements [42, 76]. Data-driven machine learning models inherit biases in the training data and discriminate users with specific sensitive attributes, e.g., gender and races [19]. Therefore, it is critical to ensure fairness in modern machine learning. In CF, researchers have defined various fairness principles from the statistical perspective to eliminate the association between sensitive attributes and recommendation results [21]. For example, the individual fairness principle calls for that similar individuals except for different sensitive attribute values should receive similar treatments [2]. Group fairness principles require that protected groups and advantaged groups should be treated similarly [21].

Researchers in the community of CF focus on designing fairness-aware CF models [65, 67]. The significance of fairness in real-world scenarios have been discussed in detail [18, 49]. To apply fairness principles to CF algorithms, researchers formulate fairness principles as different regularization terms [4, 66]. Yao et al. design four new fairness-aware metrics (e.g., value unfairness, absolute unfairness) to measure inconsistency in different user sensitive attributes for CF based recommendation, and term these metrics as regularization in the optimization objection for fairness [75]. Apart from the regularization, fair representation learning has also gained growing attention, in which adversarial training is a widely applied technique [41, 78]. It transforms the original embedding space into a new embedding space and uses discriminators to encourage the new space containing no sensitive information [41, 69]. To remove correlations among different sensitive attributes, a composition of filters and discriminators on multiple sensitive attributes have been applied for CF [5]. Furthermore, considering that user sensitive attributes are not always available in the real world, e.g., only 14% teen users show their complete profiles on Facebook [40], FairGNN is proposed to capture the graph structure information in depth for simultaneously predicting missing sensitive values, and achieving group fairness on predicted attributes [13]. Researchers have found that very abstract problem operationalizations are prevalent in fairness-aware studies, and discussed the necessity and requirements of a fair recommendation [15]. Further, researchers have started from real applied metrics to obtain a unified fairness-aware model rather than focusing on metrics with abstract fairness definitions [1]. Though great progress has been made, these methods only optimize specific fairness metric objections to achieve fairness, in which the statistical correlation between specific sensitive attributes and predicted results is removed. However, the causal relations between sensitive attributes and predicted results are largely ignored, which may reveal the real discrimination in CF and should be paid more attention.

2.3 Causal-based User Fairness

Recently, researchers have noticed that causal relations between input and output can better interpret discrimination in prediction tasks. Based on Pearl's causal graphs, causal-based fairness-aware models have been proposed [11, 12, 14, 24, 29, 52, 71], in which counterfactual fairness plays an important role. It enforces distribution over possible predictions should remain unchanged in the counterfactual world where sensitive attributes had been different [29]. Researchers leverage the causal framework of Pearl [46], and utilize a latent proxy exogenous variable to represent user characteristics unrelated to sensitive attributes. Given a pre-defined causal graph stating the causal relationships among variables, researchers have successfully achieved causal fairness [12, 17, 33, 35, 62]. For example, Matt et.al first propose to revisit fairness issues from a causal perspective [29], and

Chiappa et al. propose path-specific counterfactual fairness to distinguish the effects of different causal paths in a causal graph, and eliminated causal effects along specific unfair causal paths [12]. Li et al. leverage adversarial learning to control the dependency between user sensitive attributes and user embeddings based on a pre-defined causal graph [33].

Despite the remarkable success, how to achieve counterfactual fairness in CF remains unsolved. As the high-dimension interactions are highly correlated with user sensitive attribute, it is hard to estimate counterfactual user interactions by finding a proxy exogenous variable to represent interactive information unrelated to the sensitive information. In this paper, we adopt the classical Rubin-Neyman Causal Framework, which has been well-studied in healthcare, economics, and so on [53, 56]. The core of the Rubin-Neyman Causal Framework is to estimate the average causal effect of a treatment variable (the sensitive attribute) on a result variable by simulating a randomized controlled trial. Along this line, we can achieve the average counterfactual fairness by minimizing the estimated average causal effects. Different estimation methods have been proposed to estimate or remove the causal effects of a treatment variable, such as matching method [56], doubly robust method [20], and Inverse Propensity Weighting (IPW) based method [55, 63]. Among all these models, IPW is one of the most suitable methods in recommender systems [55, 63]. With IPW, a separate model is built to predict propensities of whether an item had been exposed to a user with only limited exposure data for simulating a random exposure situation [34].

3 PRELIMINARY

In this section, we introduce notations for CF based recommender systems, followed by some essential notations for Rubin-Neyman potential outcome framework.

3.1 Notations for CF based Recommender Systems

Recommender systems usually involve with a user set U ($|U| = M$) and an item set V ($|V| = N$). We consider the implicit feedback of users, which is more common in recommender systems. In implicit feedback, users only implicitly express their preferences, e.g. click or purchase. User-item interaction behaviors can be denoted as $\mathbf{R} = \{r(u,v)\}_{M \times N}$. If user u has interacted with item v, $r(u,v) = 1$, otherwise, $r(u,v) = 0$. Embedding based approaches have been the default choices of most recommender systems [23, 50]. Specifically, embeddings can be represented as a learnable matrix, $\mathbf{E} = [\mathbf{E}_U, \mathbf{E}_V] = [\mathbf{e}_1, ..., \mathbf{e}_u, ..., \mathbf{e}_M, ..., \mathbf{e}_{M+N}] \in \mathbb{R}^{(M+N) \times d}$. d denotes the dimension of embeddings. $\mathbf{e}_u$ and $\mathbf{e}_v$ denote corresponding embeddings of user u and item v, respectively. The predicted preference $\hat{r}(u,v)$ is calculated as the inner dot of corresponding embeddings: $\hat{r}(u,v) = \mathbf{e}_u^T \mathbf{e}_v$. To optimize the trainable parameters $\mathbf{E}$, Bayesian Personalized Ranking (BPR) is a commonly used pair-wise based optimization function [50]:

$$\min_{\mathbf{E}} \mathcal{L}_{A42} = \sum_{u=1}^{M} \sum_{(i,j) \in \mathcal{D}_u} -\ln \sigma(\hat{r}(u,i) - \hat{r}(u,j)) + \lambda ||\mathbf{E}||^2, \tag{1}$$

where $\sigma(\cdot)$ is the sigmoid function. The training dataset is represented as $\mathcal{D}_D = \{(i,j) | i \in \mathcal{R}_D \wedge j \in V - \mathcal{R}_D\}$. $\mathcal{R}_D = \{i | r(u,i) = 1\}$ and $\mathcal{V} - R_D = \{i | r(u,i) = 1\}$ denote user u's interacted items in the training data, and user u's non-interacted items, respectively.

Following previous works on recommendation fairness [57, 75], we focus on a single sensitive attribute with binary values (e.g., gender). We leverage $\mathbf{S} = [s_1, ..., s_D, ..., s_M]$ ($s_\vartheta \in \{0, 1\}$) to denote the binary user sensitive attribute. In this way, U can be divided into three subsets: subset U_1 ($|U_1| = M_1$) whose sensitive attribute value is 1, subset U_0 ($|U_0| = M_0$) whose sensitive attribute value is 0, and subset $U_{F>B}$ ($|U_{F>B}| = M - (M_1 + M_0)$) whose sensitive attribute is missing.

3.2 Potential Outcome Framework

Causal effect estimation is the core component in causal inference, which has been studied for decades in many research areas, e.g., political science, economics, and health care [53]. It analyzes the causal effects from historical data, and provides valuable suggestions on whether to take an intervention. We start from the classical Rubin-Neyman Potential Outcome Framework. This framework has been widely used in the statistical analysis of cause and effect. Suppose there are three variables: input variable X, outcome variable Y, and treatment variable T. Based on different treatments ($T = 0, T = 1$), the potential outcome for individual l has two potential values ($Y_{l=0}(x_l), Y_{l=1}(x_l)$), abbreviated as ($Y_0(x_l), Y_1(x_l)$). The connections between observed values and potential values can be formulated as:

$$y_l = t_l Y_1(x_l) + (1 - t_l) Y_0(x_l), \tag{2}$$

where t_l is the treatment for the individual l. The individual-level causal effect of the treatment variable is the difference between the potential outcome if the individual receives the treatment ($Y_1(x_l)$) and the potential outcome if she does not ($Y_0(x_l)$). Individual Treatment Effect (ITE) can be represented as:

$$\forall l \in L, ITE = Y_1(x_l) - Y_0(x_l). \tag{3}$$

Obviously, it is impossible to simultaneously see both potential outcomes. One of the potential outcomes is always missing for each individual. This problem is known as the "fundamental problem of causal inference" [53]. Consequently, individual-level treatment effects can not be directly observed or obtained. A line of causal effect estimation is randomized experiments. Note that, randomized experiments allow for population-level causal effect estimation. Randomization requires assigning treatment values randomly to each individual. Then, an estimate of the Average Treatment Effect (ATE) can be achieved by computing the differences between the treatment samples ($T = 1$) and the control samples ($T = 0$). ATE can be formulated as:

$$ATE = \mathsf{E}_{l \sim I}[Y_1(x_l) - Y_0(x_l)], \tag{4}$$

Due to ethical or practical concerns, randomly assigning treatment (e.g., taking or not taking pills) is unreasonable in the real world. Therefore, randomized experiments are hard to realize. In this case, we can only collect non-randomly treatment-assigned data (also known as observed data) for causal inference. Researchers have developed many methods to estimate ATE with these observed data. E.g., Grouped Conditional Outcome Modeling (GCOM) [28], matching based methods [56], and propensity based methods [51]. All these methods attempt to simulate random assignment by finding control individuals similar to treatment individuals. Although these methods achieve good performance in modeling variables' relationships, it is still a challenging task to adapt them for recommendation fairness. We will introduce the details of transferring potential outcome framework for recommendation fairness in the next section.

3.3 Counterfactual Fairness Definition for Collaborative Filtering

In this section, we present the counterfactual fairness definition in CF models, followed by analyzing why traditional CF models can not meet the presented counterfactual fairness definition.

In a CF model, the observed user-item interaction $\mathsf{R} = \{r(u, v)\}^{m \times n}$ can be treated as the *input variable* X, predicted preference $\hat{r}(u, v)$ is the *output variable* Y, sensitive attributes S are regarded as the *treatment* T. Along this line, we can obtain potential predicted preferences ($\hat{r}_1(u, v), \hat{r}_0(u, v)$), similar as potential outcomes. Then, we re-build connections between potential outcomes and

counterfactual notions:

$$\begin{aligned} \hat{r}(u,v) &= s_D \cdot \hat{r}_1(u,v) + (1-s_D) \cdot \hat{r}_0(u,v), \\ \hat{r}\quad(u,v) &= s_D \cdot \hat{r}_0(u,v) + (1-s_D) \cdot \hat{r}_1(u,v), \end{aligned} \tag{5}$$

where $\hat{r}\quad(u,v)$ represents the predicted preferences from counterfactual world. Based on these notions, we propose the counterfactual fairness definition as follows:

Definition 1 (Counterfactual Fairness in Collaborative Filtering). *A CF model is counterfactually fair from the user side if it meets*

$$\forall u \in U, \forall v \in V, \hat{r}(u,v) = \hat{r}\quad(u,v). \tag{6}$$

In this paper, we borrow the success of Rubin's potential outcome framework on estimating causal effects for fairness issues. Specifically, we replace the counterfactual notion $\hat{r}\quad(u,v)$ and the real world notion $\hat{r}(u,v)$ in Definition 1 with Eq.5.

$$\hat{r}(u,v) - \hat{r}\quad(u,v) = (2s_D - 1) \cdot (\hat{r}_1(u,v) - \hat{r}_0(u,v)) = 0. \tag{7}$$

Then, the counterfactual fairness requirement can be simplified as follows:

$$\forall u \in U, \forall v \in V, \hat{r}_1(u,v) - \hat{r}_0(u,v) \rightarrow 0. \tag{8}$$

Ordinarily, sensitive attributes (e.g., gender or race) can not be changed for any individual. As a result, one of the potential preferences $(\hat{r}_1(u,v), \hat{r}_0(u,v))$ is always missing, and we can not directly obtain individual-level counterfactual fairness in Eq.6. As a substitute, we utilize randomized experiments for population-level causal effect estimation. The corresponding fairness definition can be formulated as:

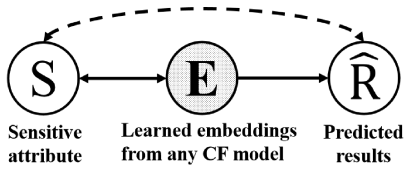
Definition 2 (Average Counterfactual Fairness in Collaborative Filtering). *A CF model satisfies the average counterfactual user fairness requirement if*

$$\begin{aligned} &\mathsf{E}_{D \bullet E \sim ?(D \bullet E)}[\hat{r}_1(u,v) - \hat{r}_0(u,v)] \\ =&\mathsf{E}_{D \bullet E \sim ?(D \bullet E)}[\hat{r}_1(u,v)] - \mathsf{E}_{D \bullet E \sim ?(D \bullet E)}[\hat{r}_0(u,v)] \rightarrow 0. \end{aligned} \tag{9}$$

In this paper, we choose to average counterfactual fairness in Eq.(9) as our goal. The detailed reasons are organized as follows: first, in our case, counterfactuals cannot be directly observed, i.e., a user cannot be both male and female at the same time. That is to say, we can not achieve Eq.(6) based on observations. Second, conducting a randomized controlled experiment is a classic approach in causal effect estimation, however, this approach still fails as we cannot conduct a randomized controlled experiment directly as we cannot randomly assign or intervene in user gender. At this point, we realize that we cannot observe or estimate Eq.(6). Third, with the help of the potential outcome framework, we find that we can estimate the overall gender effects on predicted ratings in the presence of missing values. In summary, although Eq.(6) appears to be a better goal, we can only estimate gender effects at the population level in Eq.(9). Minimizing Eq.(9) indicates that there is no gender effects on predicted ratings at the population level, which is also important for fairness-aware CF models.

However, traditional CF models can not meet the counterfactual fairness definition in Eq.9. In order to explain it, we present the process of training traditional CF models in Figure 1 a). S denotes the sensitive attribute. E denotes learnable embeddings from any CF model, and $\hat{\mathsf{R}}$ denotes predicted preferences. The relationships between these variables are analyzed as follows:

- $S \leftrightarrow \mathsf{E}$: user and item embeddings are learned from historical interactions by any CF model. Although embedding learning is not directly correlated with sensitive attributes, researchers find that users' sensitive attributes are predicted from their historical interactions [69]. Therefore, E has indirect connections with S by historical interactions.



a) The training process of traditional CF models.

Metric \ Model	MovieLens-1M		Lastfm-360K	
	BPR	GCN	BPR	GCN
Accuracy ↑	0.7413	0.7484	0.6522	0.6688
AUC ↑	0.8092	0.8133	0.6845	0.6924

b) Sensitive attribute prediction performance on user embeddings $\mathbf{E}_U$.

Fig. 1. Analyses of unfairness in traditional CF models. The left part illustrates the training process of traditional CF models, which accounts for why sensitive information are correlated with recommendation results. The right part is sensitive attribute prediction performance on two datasets. We first train two classic CF models (BPR, GCN), and then extract learned user embeddings from well-trained models. We observe that learned embeddings are helpful for sensitive attribute prediction.

- $\mathsf{E} \rightarrow \hat{\mathsf{R}}$: predicted results are transformed from corresponding user embedding and item embedding, e.g., inner dot [50] or a neural network [64].
- $S \leftrightarrow \hat{\mathsf{R}}$: according to $S \leftrightarrow \mathsf{E}$ and $\mathsf{E} \rightarrow \hat{\mathsf{R}}$, there are suspicious correlations between sensitive attributes and predicted results.

Due to existence of $S \leftrightarrow \hat{\mathsf{R}}$, we argue that the predicted results would be changed if the sensitive attribute was changed. It leads to differences in data distribution between $\hat{r}_1(u, v)$ and $\hat{r}_0(u, v)$, which contradicts the average counterfactual fairness definition in Eq.9.

To further verify the above analyses, we show an example of whether sensitive attributes are correlated with learned embeddings ($S \leftrightarrow \mathsf{E}$). We utilize a MLP based classifier to learn the mapping function from learned embeddings to the sensitive attribute. Specifically, we first split users into training/testing with the ratio of 8:2. Then, we train the classifier from the 80% users', and test classification performance on the remaining 20% users. Better performance denotes tighter correlations between embeddings and the sensitive attribute. Note that, the experiments are conducted on two widely-used recommendation datasets, and learned embeddings are from two classic CF models, i.e., BPR [50] and GCN [32]. As shown in Figure 1 b), the mapping function from E to S can be easily learned, which can be evidence of the existence of the vague correlations between S and E.

4 THE PROPOSED MODEL

In this section, we present the details of our proposed *CFFair*. As illustrated in Figure 2, inspired by the Rubin-Neyman potential outcome framework, we first transfer "counterfactual fairness for recommendation" to the "sensitive attribute's causal effect estimation" process. In the following, we introduce how to formulate the fairness goal as an additional regularization term to basic CF models with propensity scores (Section 5.1). Note that, the key of the additional regularization term lies in high-quality propensity scores. To ensure the high quality of propensity scores, we design a novel propensity estimator, which utilizes self-supervised learning to improve performance of propensity estimations (Section 5.2). Finally, we provide a model discussion in detail to clarify some default choices of our proposed *CFFair* (Section 5.3).

4.1 Average Counterfactual Fairness Formulation with Propensity Scores

In this part, we focus on how to formulate and achieve average counterfactual fairness in CF models. First, we introduce how to use observational data for estimating potential preferences. Second, we

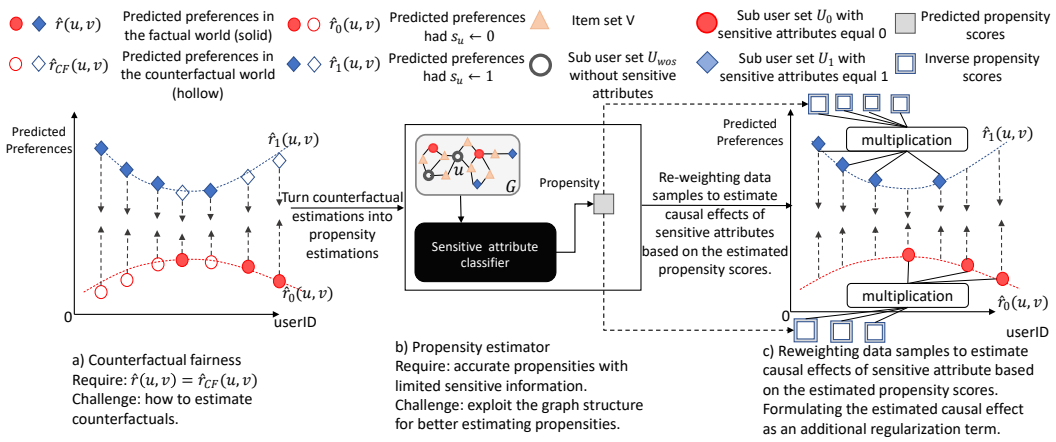


Fig. 2. The overall process of our proposed *CFFair*. a) This part illustrates the counterfactual fairness requirement in recommender systems. b) To accurately estimate propensities, we propose a novel graph based propensity estimator. c) We use inverse propensities for data sample re-weighting. In this way, we estimate the sensitive attribute's causal effects and formulate the causal effects as an additional regularization term. The size of inverse propensity is proportional to its value. The larger values of propensities correspond to the smaller weights for the data samples.

introduce how to formulate average counterfactual fairness based on potential preferences. Third, we introduce a regularization term to achieve the formulated counterfactual fairness goal.

4.1.1 Estimating potential preferences. As shown in Definition 2, the most critical part of average counterfactual fairness lies in estimating potential preferences, i.e., $\mathsf{E}_{D \bullet E \sim ?(D \bullet E)}[\hat{r}_1(u,v)]$ and $\mathsf{E}_{D \bullet E \sim ?(D \bullet E)}[\hat{r}_0(u,v)]$. Note that, there are many methods to estimate the causal effects, e.g., Grouped Conditional Outcome Modeling (GCOM) [28], matching-based methods [56], propensity-based methods [51], doubly robust based methods [20], and so on. We argue that propensity based methods are the most suitable method for the average counterfactual fairness in recommender systems. The detailed discussion of comparison among these methods is recorded in Section 4.3.1. Here, we directly leverage propensity scores, and focus on introducing how to achieve the estimation from user inherent characteristics.

For simplicity and convenience of understanding, we start with the estimation of potential preferences $\hat{r}_1(u,v_1)$ and $\hat{r}_0(u,v_1)$ with a single item v_1. Here, we take the estimation of $\hat{r}_1(u,v_1)$ as an example. $\hat{r}_1(u,v_1)$ can be divided into two parts. One part is $\hat{r}_1(u,v_1)$ for user group U_1, which can be directly estimated by any CF model. However, the other part is $\hat{r}_1(u,v_1)$ for user group U_0, which are impossible to directly obtain. As a result, the estimation of $\hat{r}_1(u,v_1)$ consists of the real-world part and the counterfactual-world part. To solve the problem, we take Bayes' theorem into consideration and design an alternative way. First, we sample from $p(u,v_1|s_D = 1)$ to avoid directly computing counterfactuals:

$$p(u,v_1) = \frac{p(u,v_1|s_D = 1)p(s_D = 1)}{p(s_D = 1|u,v_1)}. \tag{10}$$

In Eq.(10), we successfully build connections between observational $p(u,v_1|s_D = 1)$ and potential $p(u,v_1)$ for estimating $\hat{r}_1(u,v_1)$. Therefore, we can turn the estimation of potential preferences

$\hat{r}_1(u, v_1)$ as a statistical estimation as follows:

$$\begin{aligned}
&\mathsf{E}_{D \bullet E_1 \sim ?(D \bullet E_1)}[\hat{r}_1(u, v_1)] \\
=&\mathsf{E}_{D \bullet E_1 \sim ?(D \bullet E_1 | B_u = 1)}[\frac{p(s_D = 1)}{p(s_D = 1|u, v_1)} \hat{r}(u, v_1)] \\
=&\frac{1}{M_1} \tilde{\sum}_{D \in {}^*{}_1} [\frac{M_1/(M_1 + M_0)}{p(s_D = 1|u, v_1)} \hat{r}(u, v_1)] \\
=&\frac{1}{(M_1 + M_0)} \tilde{\sum}_{D \in {}^*{}_1} [\frac{\hat{r}(u, v_1)}{p(s_D = 1|u, v_1)}],
\end{aligned} \tag{11}$$

where $p(u, v_1|s_D = 1)$ corresponds to predicted preferences between group U_1 and item v_1. The probability $p(s_D = 1)$ can be directly obtained from the proportion of users U_1, denoted as $p(s_D = 1) = M_1/(M_0 + M_1)$. As for the estimation of $p(s_D = 1|u, v_1)$, we leave the solution in Section 4.2.

Meanwhile, we can leverage the similar method to estimate $\mathsf{E}_{D \bullet E_1 \sim ?(D \bullet E_1)}[\hat{r}_0(u, v_1)]$. The process can be formulated as:

$$\mathsf{E}_{D \bullet E_1 \sim ?(D \bullet E_1)}[\hat{r}_0(u, v_1)] = \frac{1}{(M_1 + M_0)} \tilde{\sum}_{D \in {}^*{}_0} [\frac{\hat{r}(u, v_1)}{p(s_D = 0|u, v_1)}]. \tag{12}$$

4.1.2 Formulating average counterfactual fairness goal. Based on the estimation of potential preferences $\hat{r}_1(u, v_1)$ and $\hat{r}_0(u, v_1)$, we can easily formulate the average counterfactual fairness with a single item v_1 as follows:

$$\frac{1}{M_0 + M_1} [\tilde{\sum}_{D \in {}^*{}_1} [\frac{\hat{r}(u, v_1)}{p(s_D = 1|u, v_1)}] - \tilde{\sum}_{D \in {}^*{}_0} [\frac{\hat{r}(u, v_1)}{p(s_D = 0|u, v_1)}]] \to 0. \tag{13}$$

Eq.13 only focuses on counterfactual fairness issues for a single item. To ensure the overall fairness of a CF model, we need to extend the requirement (Eq.13) from a single item to all items. However, the requirement of all items means that we need to consider the whole predicted dense matrix when training a CF model, which apparently leads to the increased complexity and memory requirements[1]. This prompts us to consider which items should be considered in the fairness regularization. We find that a large number of ratings for unpopular long-tailed items do not significantly contribute to the fairness goal. Therefore, we design the sampling methodology. The sampling process consists of two steps. In the first step, we filter out products with click counts below a certain threshold to exclude unpopular items. In the second step, we measure biases between males and females within these popular items and select a few items with the highest biases as the sampled results. In summary, we propose to sample important items that have noticeable differences between U_0 and U_1 to satisfy average counterfactual fairness requirements, which can ensure the fairness performance and convenience at the same time. The process can be formulated as:

$$\frac{\sum_{E \in +} 1_{X_{v\,i\,X'}} [\sum_{D \in {}^*{}_1} [\frac{\hat{A}(D \bullet E)}{?(B_u = 1|D \bullet E)}] - \sum_{D \in {}^*{}_0} [\frac{\hat{A}(D \bullet E)}{?(B_u = 0|D \bullet E)}]]}{(M_0 + M_1) \sum_{E \in +} 1_{X_{v\,i\,X'}}} \to 0, \tag{14}$$

where δ_E denotes item v's differences between U_0 and U_1. E.g., in recommender systems with explicit rating values, δ_E can be calculated as the average rating differences between U_0 and U_1. For implicit feedback, for each item v, we calculate the difference between the percentage of U_0 that rated item v and the percentage of U_1 that rated item v. $1_{X_{v\,i\,X'}}$ denotes an indicator function. If the

[1]During the actual training process, we found that using all the data for fairness-aware regularization term calculation on the Lastfm-360K dataset exceeded the memory capacity of our 3090 GPU with 24G memory.

requirement is met (i.e., $\delta_E > \delta'$), the indicator equals 1. Otherwise, $1_{X_v\, i\, X'} = 0$. Please note that, we also include the sampling methodology in the baseline DP-BPR and DP-GCN.

4.1.3 Overall loss function. To this end, our goal is two-fold: maintaining recommendation accuracy (Eq.1), and meeting the average counterfactual fairness requirements (Eq.14). To achieve the two-fold goal, we propose to augment the learning objective by adding an additional fairness-aware regularization term, similar to [75]. The additional term is formulated as:

$$\mathcal{L}_{A46} = \frac{\sum_{E \in +} 1_{X_v\, i\, X'} \left[\sum_{D \in *_1} \left[\frac{\hat{A}(D \bullet E)}{\hat{?}(B_u = 1 | D \bullet E)} \right] - \sum_{D \in *_0} \left[\frac{\hat{A}(D \bullet E)}{\hat{?}(B_u = 0 | D \bullet E)} \right] \right]}{(M_0 + M_1) \sum_{E \in +} 1_{X_v\, i\, X'}}. \tag{15}$$

Correspondingly, the overall loss function for the two-fold goal is formulated as:

$$\min_{\mathsf{E}} \mathcal{L}_{0;;} = \mathcal{L}_{A42} + \mu \mathcal{L}_{A46}, \tag{16}$$

where the first loss $\mathcal{L}_{A42}$ models recommendation accuracy, and the second term is the fairness-aware regularization term. μ is a hyper-parameter to control the balance between accuracy and average counterfactual fairness performance. The detailed algorithm can be found in Algorithm 1.

Algorithm 1 Detailed training procedures of *CFFair*.

Require: Users U, items V, interactions R, estimated propensity scores $p(s_D = 1|u)$ and $p(s_D = 0|u)$ from the propensity estimator.

Ensure:

1: Initialize trainable parameters E.
2: **repeat**
3: Sample negative triplets (u, i, j) from $u \in U \; \forall\, i \in \mathcal{R}_D \; \forall\, j \in V - \mathcal{R}_D$,
4: Get a batch of training data, including triplets (u, i, j)
5: **for** each (u, i, j) in the batch **do**
6: Compute recommender system loss $\mathcal{L}_{A42}$ (Eq.1),
7: Compute the regularization term $\mathcal{L}_{A46}$ (Eq.14),
8: Compute the overall loss $\mathcal{L}_{0;;}$ (Eq.16),
9: **end for**
10: Minimize $\mathcal{L}_{0;;}$ (Eq.16) to optimize E.
11: **until** Convergence of the recommender system.

4.2 The Propensity Estimator

The remaining problem in Eq.15 is $p(s_D = 1|u, v)$ estimation. As items are irrelevant with user sensitive attributes, the estimation can be degraded to estimate $p(s_D = 1|u)$. Note that, well-trained user embeddings have been proven to include ample user inherent characteristics [5]. However, in the process of training *CFFair*, the embeddings are encouraged not to contain sensitive information, which is harmful for sensitive attribute prediction. Therefore, it is improper to estimate propensity scores based on the embeddings trained from *CFFair*. As a substitute, we use pre-trained historical user embeddings from other basic CF models, e.g., BPR [50] and GCN [23], which are considered as containing ample sensitive information. Apart from historical user embeddings, the unique user-item bipartite graph structure in recommender systems has been proven to include ample user information [57, 69]. Intuitively, we further propose to utilize structure information to improve propensity estimation performances.

To sum up, we treat the process of propensity estimation as a classification task. The input of the classification task is the historical user embeddings. The predicted probability of $(s_D = 1)$ can

be seen as propensities $p(s_D = s|u)$. To this end, we develop a propensity estimator, consisting of a graph self-supervised encoder for embedding learning and a sensitive attribute classifier for propensity estimation. The overall structure of the estimator is shown in Figure 3.

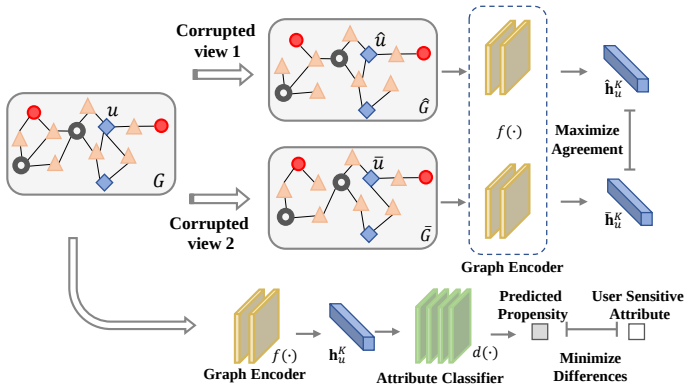


Fig. 3. The structure of the propensity estimator.

4.2.1 Graph self-supervised encoder $f(\cdot)$. The encoder transforms historical embeddings to graph self-superivsed representations, which encourages encoding more sensitive information. The encoder is shown in the yellow module in Figure 3.

In recommender systems, user-item behavior data naturally forms a user-item bipartite graph. Empirically, graph-based methods can capture high-order collaborative signals, obtain more effective embeddings, and have better recommendation results [23, 32]. Hence, we intend to utilize the bipartite graph structure for propensity estimation. Besides, self-supervised signals (i.e., user attributes) have been proven effective in improving effectiveness of classification performance [77]. To this end, we focus on applying the utilization of the user-item bipartite graph and self-supervised learning techniques into the propensity estimation process.

To utilize the graph structure information, we employ Graph Convolutional Network (GCN) [26] to aggregate neighboring nodes' features according to the user-item bipartite graph. Specifically, the aggregation process at the $(k+1)$ iteration can be formulated as the following two steps:

$$\begin{aligned} \mathbf{h}_E^{k+1} &= GCN(\mathbf{h}_E^k, \{\mathbf{h}_D^k : u \in \mathcal{R}_E\}), \\ \mathbf{h}_D^{k+1} &= GCN(\mathbf{h}_D^k, \{\mathbf{h}_E^k : v \in \mathcal{R}_D\}), \end{aligned} \tag{17}$$

where $\mathcal{R}_D$ denotes neighboring item nodes of user u (aka user u's interacted items). Correspondingly, $\mathcal{R}_E$ denotes neighboring user nodes of item v. $\mathbf{h}_D^0, \mathbf{h}_E^0$ denotes historical user representations and item representations, respectively. Please refer to Section 5.1 for details about $\mathbf{h}_D^0, \mathbf{h}_E^0$.

Note that, supervised signals (i.e., labels of user attributes) are necessary for propensity estimations. However, users are not willing to expose their sensitive information in the real world, which poses a great challenge of lacking effective supervised signals. As self-supervised signals (i.e., user attributes) have been proven effective in improving effectiveness of classification performance [77], we propose to employ self-supervised learning techniques to alleviate the challenge.

In detail, we first generate two corrupted graphs (i.e., $\hat{G}_1$ and $\bar{G}_2$) by randomly dropping graph nodes in the bipartite graph. Then, we leverage Eq.17 to process two corrupted graphs and obtain

representations from the last layer of graphs, which are denoted as

$$\begin{aligned} \hat{\mathbf{h}}_D &= f(\mathbf{h}_D^0)\ for\ \hat{G}_1, \\ \bar{\mathbf{h}}_D &= f(\mathbf{h}_D^0)\ for\ \bar{G}_2. \end{aligned} \tag{18}$$

Next, the self-supervised signals are captured by maximizing the agreement between representations of the same user u in different corrupted graphs (i.e., $\hat{G}_1$ and $\bar{G}_2$) for better embedding learning. The intuition behind this operation is that robust representations should hold unchanged under different kinds of graph disturbances. The process of maximizing agreement can be formulated as:

$$\begin{aligned} \max_{f} \mathcal{L}_{BB;} &= \tilde{O}_{D=1} \ln \frac{c(\hat{\mathbf{h}}_D, \bar{\mathbf{h}}_D)}{c(\hat{\mathbf{h}}_D, \bar{\mathbf{h}}_D) + \ _D}, \\ _D &= \tilde{O}_{<=1} 1_{[<<D]}(c(\hat{\mathbf{h}}_<, \bar{\mathbf{h}}_D) + c(\hat{\mathbf{h}}_D, \bar{\mathbf{h}}_<)), \\ c(\hat{\mathbf{h}}_D, \bar{\mathbf{h}}_D) &= e^{\vartheta(\hat{\mathbf{h}}_u^K \cdot \bar{\mathbf{h}}_u^K)}, \end{aligned} \tag{19}$$

where ϑ denotes cosine similarity. $e^{\vartheta(\hat{\mathbf{h}}_u^K \cdot \bar{\mathbf{h}}_u^K)}$ denotes the similarity between the same user u's two vectors $\hat{\mathbf{h}}_D$ and $\bar{\mathbf{h}}_D$ based on two corrupted graphs $\hat{G}_1$ and $\bar{G}_2$. Note that, $1_{[<<D]}$ is an indicator function. If $[m < u]$ is true, $1_{[<<D]} = 1$, otherwise, $1_{[<<D]} = 0$. $_D$ denotes the similarity between different users' vectors based on two corrupted graphs. The optimization parameters $_5$ denotes the trainable parameters of encoder $f(\cdot)$.

4.2.2 Sensitive a ribute classifier. The classifier transforms graph self-superivsed representations to propensity scores, which encourages better classification performance. The classifier is shown in the green module in Figure 3.

As mentioned before, we treat the propensity estimation $p(s_D = 1|u, v)$ as a sensitive attribute classification task. As a result, we leverage a classifier $d(\cdot)$, such as a Multi-Layer Perceptron (MLP) [43], to achieve this goal, which can be formulated as follows:

$$p(s_D = 1|u) = d(\mathbf{h}_D). \tag{20}$$

We leverage the Cross-Entropy function for optimization:

$$\min_{f^{\bullet}\ d} \mathcal{L}_{2;B} = \tilde{O}_{D \in *_1 \cup *_0} -[s_D \ln p(s_D = 1|u) + (1 - s_D) \ln(1 - p(s_D = 1|u))]. \tag{21}$$

Then, the overall optimization target of our proposed propensity estimator is formulated as follows:

$$\min_{f^{\bullet}\ d} \mathcal{L}_{2;B} - \beta \mathcal{L}_{BB;}, \tag{22}$$

where β is a hyper-parameter that controls weight for self-supervised loss. After training the whole propensity estimator, we can obtain all propensity scores from Eq.20. To overcome high variances of estimated propensity scores, we introduce Clipped Propensity Score [54] as follows:

$$p(s_D = s|u) = max\{p(s_D = s|u), \rho\}, s \in \{0, 1\}, \tag{23}$$

where ρ is a hyper-parameter that controls scope of "Clipped". This process has been reported in Algorithm 2. By employing this estimator, we can obtain the propensity scores for the final fair recommendation.

Algorithm 2 User sensitive attribute propensity estimation.

Require: Users U, Item V, users with sensitive attributes $U_1 \cup U_0$, sensitive attributes S.
Ensure:
Initialize trainable parameters $_5$, $_3$ of propensity estimator.
repeat
  Get a batch from users $U_1 \cup U_0$.
  for each (u, s_D) in the batch **do**
    Get user u's aggregated representations h_D (Eq.17),
    Compute self-supervised learning loss $\mathcal{L}_{BB;}$ (Eq.19),
    Compute semi-supervised classification loss $\mathcal{L}_{2;B}$ (Eq.21),
  end for
  Compute and minimize Eq.22 to optimize $_5$, $_3$,
until Convergence of the estimator.
Output:
Predicted propensity scores $p(s_D = s|u, v)$.

4.3 Model Discussion

4.3.1 Comparisons among di erent fairness definitions. Counterfactual fairness [12, 29, 71] is closer to individual fairness [2, 45, 48]. Specifically, both definitions aim to eliminate differences between two individuals. Individual fairness requires that similar individuals with different sensitive attributes receive similar outcomes. Counterfactual fairness, on the other hand, defines two different worlds: the factual world and the counterfactual world. In the counterfactual world, each user's gender is changed. Unlike individual fairness, which focuses on reducing differences between similar real users, counterfactual fairness aims to eliminate differences between each real individual and their corresponding virtual individual in the counterfactual world.

Average counterfactual fairness is more aligned with a group fairness definition, demographic parity [38]. In contrast to removing the influence of sensitive attributes on outcomes from a causal perspective, demographic parity focuses on mitigating existing biases in observed data. We have to acknowledge that our defined average counterfactual fairness goal in Eq.9 can be easily reduced to demographic parity in two cases. One case is that historical user embeddings are totally irrelevant to the sensitive attribute. In this case, the predicted propensity scores tend to be around the classification threshold. We argue that the threshold for an imbalanced binary classification is closer to the category with more instances [6]. In our task, the threshold of sensitive attribute classification should be $\frac{"_1}{"_1+"_0}$ for U_1, and $\frac{"_0}{"_1+"_0}$ for U_0, respectively. To this end, average counterfactual fairness will degenerate to $\mathsf{E}_{D\bullet E_1\sim?(D\bullet E_1|B_u=1)}[\hat{r}(u, v_1)] - \mathsf{E}_{D\bullet E_1\sim?(D\bullet E_1|B_u=0)}[\hat{r}(u, v_1)]$ (demographic parity). In addition, the poor performance of the classifier module can make predicted propensity scores tend to be around the classification threshold, leading to the degeneration. Therefore, it is also necessary to improve classification performance. For example, we choose to apply the graph self-supervised estimator to obtain accurate sensitive attribute values.

4.3.2 Comparisons to other causal e ect estimation methods. There are many other casual effect estimation methods, e.g., Grouped Conditional Outcome Modeling (GCOM) [28], matching based methods [56], doubly robust based methods [20], and so on.

We argue that propensity based methods are the most suitable method for the average counterfactual fairness in recommender systems, and other causal effect estimation methods are not currently suitable. Specifically, GCOM builds two different models for different sensitive attribute values, e.g., two CF models respectively for male users and female users. Then, GCOM can estimate

a male/female user's counterfactual results by inputting the user information to female/male CF models. The success of GCOM lies in using other associated information to build two models and exchanging the input of these two models. However, this fails in recommender systems. Suppose that two CF models are designed for only male users and female users. Then, when exchanging input of two CF models, exchanged users are new users for each CF model. Note that, "new user" problem (i.e., cold start problem) is a common challenge for CF models, and it is almost impossible to give proper prediction without additional side information in CF models. Therefore, GCOM fails in CF based recommender systems.

Matching-based methods are not very suitable in CF models. In counterfactual fairness in recommender systems, matching-based methods should first divide users into male users and female users. Then, they should match users in different user groups, i.e., matching a most similar male/female user for a female/male user. When it comes to causal effect estimation, matching-based methods would compare each user and the matched user. However, this process is difficult in recommender systems, as it is very hard to match "similar users" with high-dimension user characteristics (i.e., sparse user behaviors). Doubly robust based methods are not fit for fairness in recommendation, as matching or GCOM are base models for doubly robust based methods.

Luckily, propensity scores based methods can successfully measure average counterfactual fairness in recommender systems. Propensity scores based methods utilize high-dimension user characteristics to first estimate a propensity score for each sample, and then reweight training samples with inverse propensity scores for causal effect estimation. Note that, propensity scores based methods avoid directly estimating counterfactuals and are suitable for high-dimension unobserved latent interest. Due to the above analyses, we argue that propensity score based methods are currently the most suitable for counterfactual fairness estimation in CF models. Therefore, we leverage propensity scores to achieve the causal effect estimation.

5 EXPERIMENTS

In this section, we first introduce three real-world datasets that we evaluate models. Then, we describe baseline models, evaluation metrics, and the implementation details of our proposed *CFFair*. Next, we present empirical results and give a detailed analysis of models on recommendation performance and fairness performance.

5.1 Experimental Settings

Table 1. Statistics of the three datasets.

Datasets	Users	Items	Ratings	Density
PISA-Australia	8,476	184	93,571	6.000%
MovieLens-1M	6,038	3,533	575,281	2.697%
Lastfm-360K	53,675	125,512	2,621,895	0.039%

Datasets. We select *PISA-Australia*[2] dataset, *MovieLens-1M* [22] dataset and *Lastfm-360K* dataset [8], whose statistics are reported in Table 1. In order to turn *MovieLens-1M* into an implicit feedback dataset, we treat samples with rating of 4 and 5 as positive feedback and the rest as negative feedback, similar as previous studies did [80]. We split the training/validation/testing sets with the ratio of 70%, 10%, 20% on all datasets. Meanwhile, we treat *gender* as the sensitive attribute. The male users are treated as the user group U_1. In constrast, female users are regarded as U_0. This operation is similar as many previous works did on fairness [75]. To simulate the situation that user

[2]https://www.oecd.org/pisa/data/

may not expose their sensitive information in real world, we randomly drop 70% users' sensitive attributes and only keep 30% users' sensitive attributes when training the propensity estimator.

The *PISA-Australia* dataset consists of students' responses logs on different exercises, which can be seen as students' behaviors on exercises. We treat students as users, exercises as items, and our goal is to predict students' scores on exercises. In this way, we can recommend their non-mastered exercises to students. To transform the PISA-Australia dataset into implicit feedback, we retain students' correct response logs as positive interactions, and remove users with less than 12 behaviors. We have discovered an obvious disparity in the proportion of correctly answers for most exercises between males and females. Among 184 exercises (items), over 20% of the exercises (37/184) exhibit a proportion difference of more than 5%, while over 7% of the exercises (13/184) show a proportion difference of more than 10%. Also, the student embeddings mined from these response logs can be seen as student abilities, potentially affecting subsequent decisions, such as class assignment and admission. This reminds us that we should pursue fairness on this dataset.

Baselines. We compare our proposed *CFFair* with the following baselines:

- BPR [50]: it is a classic latent factor-based CF model with pair-wise optimization.
- GCN [32]: it is a state-of-the-art graph-based CF model. It enhances recommendation accuracy by removing non-linearities and proposing a residual structure.
- DP-BPR, DP-GCN [75]: it proposes a practical statistical fairness regularization term in explicit feedback. We modify the regularization term to implicit feedback. Then, we add the modified regularization term to BPR and GCN, respectively.
- FairGo [69]: it adopts adversarial training for group fairness in CF based recommender systems. Specifically, it removes unfairness from the perspectives of both local representations and sub user graph representations. As it requires full access to sensitive attributes, we adopt the simplest method, i.e., filling missing values with the majority value of the sensitive attribute in our setting.
- FairGNN [13]: it is designed for group fairness on a graph structure with limited sensitive information. Adversarial training is adopted to remove unfairness, and a graph-based sensitive attribute classifier is utilized to handle missing sensitive information.

Evaluation metrics. For recommendation performance, we use two commonly used metrics, Hit Ratio (HR) and Normalized Discounted Cummulative Gain (NDCG) [23]. HR measures the percentage of hit items, and NDCG puts more emphasis on the top-ranked items. For counterfactual fairness performance, an intuitive method is to use the average counterfactual regularization term in Eq.14 as a metric $ATE_{\%}$. Please note that, the propensity scores used in the metric $ATE_{\%}$ are independent of all training models. $ATE_{\%}$ is formulated as:

$$ATE_{\%} = |\mathsf{E}_{D\bullet\mathsf{E}\sim?(D\bullet\mathsf{E})}[\hat{r}_1(u,v) - \hat{r}_0(u,v)]|. \tag{24}$$

The smaller $ATE_{\%}$ denotes the better average counterfactual fairness. As there is no golden standard to evaluate counterfactual fairness with observational data, our intuition is to employ more causal effect estimation methods for counterfactual fairness evaluation. Specifically, we further adopt another commonly-used matching-based method [56]. The basic idea is to match similar users but with different sensitive attributes, and then use the matched users to estimate the causal effects. For those users who match other users, we use $(u, u_<)$ pairs to denote the matching user pair. $u_<$ denotes the matching user for user u, which can be seen as the counterfactual estimation of u. The process of matching users is as follows. First, we calculate similarities among users. Second, for each user u, we find the most similar user $u_<$ with the other sensitive attribute value. Note that, we choose K:1 matching considering feasibility. If there are many similar users, with the majority being males and only a small portion being females. During the matching process, if we strictly allow 1:1

Table 2. Average counterfactual user fairness performance on MovieLens-1M with varying K.

Model	$ATE_{\%}\downarrow$	$Matching\downarrow$	M@K↑		
			K=20	K=30	K=40
BPR	1.0255 ± 0.0530	0.2120 ± 0.0236	0.0855 ± 0.0026	0.1073 ± 0.0032	0.1238 ± 0.0201
DP-BPR	0.6278 ± 0.0378	0.1428 ± 0.0225	0.1364 ± 0.0112	0.1633 ± 0.0088	0.1855 ± 0.0070
CFFair-BPR	**0.4753 ± 0.0453**	**0.1193 ± 0.0102**	**0.1398 ± 0.0029**	**0.1646 ± 0.0036**	**0.1855 ± 0.0127**
GCN	1.4403 ± 0.0560	0.3513 ± 0.0313	0.0873 ± 0.0015	0.1058 ± 0.0025	0.1198 ± 0.0036
FairGo	1.3682 ± 0.0317	0.8920 ± 0.0212	0.1053 ± 0.0164	0.1243 ± 0.0187	0.1404 ± 0.0086
FairGNN	1.0827 ± 0.0220	0.8620 ± 0.0117	0.1281 ± 0.0234	0.1525 ± 0.0293	0.1677 ± 0.0079
DP-GCN	1.1513 ± 0.0683	0.1640 ± 0.0230	0.1287 ± 0.0174	0.1478 ± 0.0170	0.1643 ± 0.0192
CFFair-GCN	**0.7552 ± 0.0271**	**0.1542 ± 0.0095**	**0.1402 ± 0.0080**	**0.1554 ± 0.0061**	**0.1687 ± 0.0053**

matching between males and females, we would find that most males do not have suitable matches, which is unacceptable. Third, to avoid poor matches in K:1 matching, we set a similarity threshold τ to filter out certain unreliable matches. In our experiments, we set $\tau = 0.6$. After obtaining matched user pairs $(u, u_<)$, we propose *Matching* to measure differences between matched pairs:

$$Matching = \mathsf{E}^{D_m}_{D\bullet E\sim ?(D\bullet E)}[\hat{r}(u,v) - \hat{r}(u_<,v)], \tag{25}$$

where the smaller *Matching* value denotes better average counterfactual fairness performance. Furthermore, we extend the matching-based method to ranking. Specifically, we propose a matching-based ranking metric by comparing the recommended lists of matched users (*Matching*@K, $M@K$):

$$M@K = \tilde{\mathsf{O}}_{D\bullet D_m} \frac{|\mathcal{L}(u)@K \cap \mathcal{L}(u_<)@K|}{K}, \tag{26}$$

where $L_D@K$ denotes items in the Top-K list for user u. Obviously, $M@K$ measures the similarity of recommended lists between matched users. The larger $M@K$ denotes more similarity between matched users, leading to a better average counterfactual user fairness.

Parameter settings. We conduct all experiments with Pytorch-1.6.0 on 1 NVIDIA TITAN RTX Graphics card. We utilize Adam optimizer and the initial learning rate is 0.005. In each epoch, we generate the training dataset $\mathcal{D}$ by randomly sampling one non-interacted item as negative item j for each user-item pair (u, i). The training dataset is filled with triplets (u, i, j) by random selecting non-interacted items j. In one batch, we random select 16,384 triplets in *MovieLens-1M* and 32,768 triplets in *Lastfm-360K* to update user and item embeddings. The embedding size is set to 64, i.e., $D = 64$. As for recommendation model training, hyper-parameter μ in Eq.16 is selected from $\{1\times10^{-5}, 2\times10^{-5}, 5\times10^{-5}, 1\times10^{-4}, 0.001, 0.01\}$ for *CFFair*-BPR, and $\{2\times10^{-6}, 5\times10^{-6}, 1\times10^{-5}, 2\times10^{-5}, 5\times10^{-5}, 1\times10^{-4}\}$ for *CFFair*-GCN. The balancing parameter λ in Eq.16 is set to 0.01. Clipped hyper-parameter ρ in Eq.23 is set to 0.1. We choose 50 items in *PISA-Australia*, items in *MovieLens-1M*, and 5,000 items in *Lastfm-360K* to calculate the regularization term in Eq.15.

As for the self-supervised learning-based propensity estimator, we randomly drop 40% user nodes to generate corrupted graphs. We obtain $\mathsf{h}^0_D, \mathsf{h}^0_E$ from a well-trained CF model. The encoder module of the propensity estimator is realized by a combination of aggregation function in Eq.17 and a 2-layer MLP. The attribute classifier module is realized by a 4-layer MLP on MovieLens-1M and 6-layer MLP on Lastfm-360K. β in Eq.22 is selected from {0.001,0.002,0.005,0.01,0.1}.

5.2 Overall Performance

Tables 5, 2, 6, and 3 report the overall results. We have several observations from these tables.

Table 3. Average counterfactual user fairness performance on Lastfm-360K with varying K.

Model	$ATE_{\%}$ ↓	*Matching* ↓	M@K↑		
			K=20	K=30	K=40
BPR	1.2680 ± 0.0223	2.0428 ± 0.0219	0.1261 ± 0.0130	0.1480 ± 0.0124	0.1661 ± 0.0241
DP-BPR	0.6140 ± 0.0179	1.8840 ± 0.0246	0.2032 ± 0.0156	0.2341 ± 0.0348	0.2571 ± 0.0281
CFFair-BPR	**0.3350 ± 0.0169**	**1.4575 ± 0.0241**	**0.2054 ± 0.0192**	**0.2342 ± 0.0256**	**0.2562 ± 0.0178**
GCN	1.4655 ± 0.0235	3.0017 ± 0.0173	0.0784 ± 0.0323	0.0952 ± 0.0136	0.0860 ± 0.0149
FairGo	1.3933 ± 0.0367	2.0765 ± 0.0488	0.1465 ± 0.0410	0.1632 ± 0.0352	0.1772 ± 0.0527
FairGNN	1.2146 ± 0.0412	2.1953 ± 0.0696	0.1476 ± 0.0325	0.1689 ± 0.0414	0.1852 ± 0.0213
DP-GCN	1.4252 ± 0.0241	2.4978 ± 0.0257	0.2466 ± 0.0198	0.2669 ± 0.0144	0.2816 ± 0.0183
CFFair-GCN	**1.0529 ± 0.0102**	**2.0960 ± 0.0178**	**0.2485 ± 0.0205**	**0.2751 ± 0.0174**	**0.2910 ± 0.0172**

Table 4. Average counterfactual user fairness performance on the PISA-Australia dataset with varying K.

Model	$ATE_{\%}$ ↓	*Matching* ↓	M@K↑		
			K=20	K=30	K=40
BPR	0.3329 ± 0.0175	1.9044 ± 0.0352	0.2894 ± 0.0146	0.3556 ± 0.0205	0.4263 ± 0.0197
DP-BPR	0.0061 ± 0.0015	0.2762 ± 0.0088	0.5072 ± 0.0233	0.6146 ± 0.0150	0.6982 ± 0.0249
CFFair-BPR	**0.0022 ± 0.0006**	**0.1843 ± 0.0124**	**0.5152 ± 0.0152**	**0.6525 ± 0.0147**	**0.7210 ± 0.0182**
GCN	0.3444 ± 0.0122	0.8936 ± 0.0244	0.4841 ± 0.0153	0.5880 ± 0.0141	0.6573 ± 0.0096
FairGo	0.0063 ± 0.0010	0.3954 ± 0.0102	0.5021 ± 0.0262	0.6124 ± 0.0275	0.6832 ± 0.0343
FairGNN	0.0046 ± 0.0012	0.3443 ± 0.0155	0.5110 ± 0.0278	0.6450 ± 0.0317	0.6956 ± 0.0290
DP-GCN	0.0057 ± 0.0010	0.3728 ± 0.0134	0.5077 ± 0.0144	0.6234 ± 0.0102	0.6905 ± 0.0083
CFFair-GCN	**0.0031 ± 0.0012**	**0.3178 ± 0.0193**	**0.5181 ± 0.0109**	**0.6549 ± 0.0081**	**0.7019 ± 0.0162**

- **First**, we observe obvious trade-off effects between fairness and accuracy for all fairness-aware models. Specifically, all fairness-aware models have better fairness performance ($ATE_{\%}$, *Matching*, M@K) than basic CF models but also suffer a decrease in recommendation performance (HR@K, NDCG@K).
- **Second**, compared to other fairness-aware models, *CFFair* outperforms baselines on all three datasets on the average counterfactual fairness metrics ($ATE_{\%}$, *Matching*, M@K), and causes the least degradation on recommendation accuracy (HR@K, NDCG@K). The results prove that *CFFair* can achieve a good balance point between accuracy and counterfactual user fairness performance. Specifically, on the PISA-Australia dataset, regardless of whether BPR or GCN is used as the base model, our proposed *CFFair* outperforms the suboptimal models by over 30% improvement in ATE and over 7% improvement in the Matching metric.
- **Third**, FairGNN has better fairness performance and recommendation performance than FairGo. The reason is that FairGNN utilizes the graph structure to fill up missing sensitive information as guidance for adversarial training. FairGo suffers from the limitation of incomplete sensitive attributes, leading to a performance decrease.
- **Fourth**, when it comes to comparing BPR based models and GCN based models, we find that GCN based models perform obviously better on recommendation performance. The reason is that GCN based models can capture high-order collaborative signals.
- **Last but not least**, the statistical fairness oriented models (DP, FairGo, FairGNN) are designed to optimize statistical fairness metrics, but they also improve the average counterfactual user fairness compared to base models. It proves that partial overlap exists between statistical fairness and average counterfactual user fairness.

Table 5. Top-K recommendation performance on MovieLens-1M with varying K.

Model	HR@K↑			NDCG@K↑		
	K=20	K=30	K=40	K=20	K=30	K=40
BPR	**0.2988 ± 0.0038**	**0.3406 ± 0.0030**	**0.3821 ± 0.0028**	**0.2716 ± 0.0023**	**0.2856 ± 0.0019**	**0.3008 ± 0.0018**
DP-BPR	0.2725 ± 0.0033	0.3145 ± 0.0045	0.3539 ± 0.0049	0.2480 ± 0.0036	0.2620 ± 0.0034	0.2765 ± 0.0036
CFFair-BPR	0.2726 ± 0.0024	0.3147 ± 0.0027	0.3543 ± 0.0028	0.2463 ± 0.0025	0.2608 ± 0.0026	0.2754 ± 0.0026
GCN	**0.3022 ± 0.0025**	**0.3435 ± 0.0031**	**0.3844 ± 0.0028**	**0.2744 ± 0.0024**	**0.2881 ± 0.0027**	**0.3031 ± 0.0023**
FairGo	0.2811 ± 0.0135	0.3252 ± 0.0210	0.3664 ± 0.0198	0.2531 ± 0.0230	0.2680 ± 0.0174	0.2833 ± 0.0156
FairGNN	0.2766 ± 0.0210	0.3202 ± 0.0187	0.3562 ± 0.0213	0.2488 ± 0.0314	0.2634 ± 0.0172	0.2796 ± 0.0190
DP-GCN	0.2819 ± 0.0088	0.3224 ± 0.0039	0.3610 ± 0.0111	0.2590 ± 0.0070	0.2730 ± 0.0063	0.2869 ± 0.0070
CFFair-GCN	0.2818 ± 0.0024	0.3245 ± 0.0022	0.3640 ± 0.0029	0.2564 ± 0.0022	0.2733 ± 0.0022	0.2871 ± 0.0024

Table 6. Top-K recommendation performance on Lastfm-360K with varying K.

Model	HR@K↑			NDCG@K↑		
	K=20	K=30	K=40	K=20	K=30	K=40
BPR	**0.1521 ± 0.0017**	**0.1960 ± 0.0026**	**0.2316 ± 0.0031**	**0.1326 ± 0.0024**	**0.1525 ± 0.0018**	**0.1673 ± 0.0046**
DP-BPR	0.1237 ± 0.0076	0.1593 ± 0.0093	0.1898 ± 0.0088	0.1105 ± 0.0051	0.1267 ± 0.0045	0.1393 ± 0.0057
CFFair-BPR	0.1373 ± 0.0085	0.1768 ± 0.0064	0.2097 ± 0.0069	0.1216 ± 0.0061	0.1396 ± 0.0043	0.1533 ± 0.0066
GCN	**0.1597 ± 0.0063**	**0.2031 ± 0.0082**	**0.2389 ± 0.0071**	**0.1426 ± 0.0045**	**0.2389 ± 0.0054**	**0.1771 ± 0.0048**
FairGo	0.1387 ± 0.0167	0.1792 ± 0.0132	0.2123 ± 0.0183	0.1222 ± 0.0095	0.1407 ± 0.0084	0.1544 ± 0.0088
FairGNN	0.1428 ± 0.0268	0.1835 ± 0.0190	0.2178 ± 0.0131	0.1267 ± 0.0169	0.1476 ± 0.0152	0.1598 ± 0.0124
DP-GCN	0.1432 ± 0.0084	0.1857 ± 0.0101	0.2204 ± 0.0093	0.1262 ± 0.0063	0.1455 ± 0.0082	0.1599 ± 0.0076
CFFair-GCN	0.1455 ± 0.0097	0.1867 ± 0.0082	0.2203 ± 0.0089	0.1294 ± 0.0078	0.1481 ± 0.0061	0.1620 ± 0.0074

Table 7. Top-K recommendation performance on the PISA-Australia dataset with varying K.

Model	HR@K↑			NDCG@K↑		
	K=20	K=30	K=40	K=20	K=30	K=40
BPR	**0.3476 ± 0.0064**	**0.3882 ± 0.0053**	**0.4299 ± 0.0041**	**0.2428 ± 0.0040**	**0.2601 ± 0.0047**	**0.2716 ± 0.0023**
DP-BPR	0.3417 ± 0.0051	0.3829 ± 0.0046	0.4223 ± 0.0071	0.2341 ± 0.0077	0.2487 ± 0.0052	0.2630 ± 0.0060
CFFair-BPR	0.3416 ± 0.0063	0.3843 ± 0.0054	0.4228 ± 0.0088	0.2331 ± 0.0042	0.2492 ± 0.0064	0.2626 ± 0.0051
GCN	**0.3515 ± 0.0078**	**0.3939 ± 0.0064**	**0.4350 ± 0.0068**	**0.2519 ± 0.0053**	**0.2682 ± 0.0041**	**0.2827 ± 0.0056**
FairGo	0.3448 ± 0.0114	0.3898 ± 0.0127	0.4301 ± 0.0122	0.2399 ± 0.0104	0.2551 ± 0.0080	0.2684 ± 0.00108
FairGNN	0.3453 ± 0.0133	0.3916 ± 0.0120	0.4325 ± 0.0094	0.2402 ± 0.0094	0.2569 ± 0.0143	0.2711 ± 0.0082
DP-GCN	0.3422 ± 0.0086	0.3913 ± 0.0077	0.4316 ± 0.0099	0.2397 ± 0.0092	0.2559 ± 0.0080	0.2701 ± 0.0075
CFFair-GCN	0.3481 ± 0.0069	0.3921 ± 0.0045	0.4338 ± 0.0068	0.2419 ± 0.0060	0.2588 ± 0.0076	0.2729 ± 0.0061

5.3 Detailed Model Analyses

In this part, we conduct more experiments to better verify the effectiveness of our proposed *CFFair*. The following questions will be answered:

(1) Is the graph self-supervised learning estimator better than other classification models?
(2) Will *CFFair* still have a relatively good performance than basic CF models (i.e., BPR, GCN) on statistical fairness metrics?
(3) How does the balancing parameter μ in Eq.(16) affect the trade-off effects of *CFFair*?
(4) Can *CFFair* achieve stable improvements on matching-based metrics with varying the similarity threshold τ?
(5) How does *CFFair* perform if removing the sampling methodology in Eq.(14)?

The effectiveness of the graph self-supervised learning estimator. As discussed in Section 4.3.1, the causal effect estimations highly rely on the accuracy of propensity estimations. To further boost the performance of propensity estimations, we propose a novel estimator which adopts graph self-supervised learning. In Section 5.2, we only evaluate recommendation and fairness

performance rather than directly evaluating the performance of propensity estimations. In this part, we conduct additional experiments to directly verify the estimator's effectiveness.

Specifically, we compare our proposed *CFFair* with several classic classification models, i.e., Label Propagation (LP) [61], MLP, GR [3], and semi-GCN [26]. As the sensitive attribute is binary, we choose accuracy and AUC as the evaluation metrics. The results in Table 8 clearly show that our proposed propensity estimator has the best performance on three datasets. The results prove that graph self-supervised learning is effective to boost estimation performance.

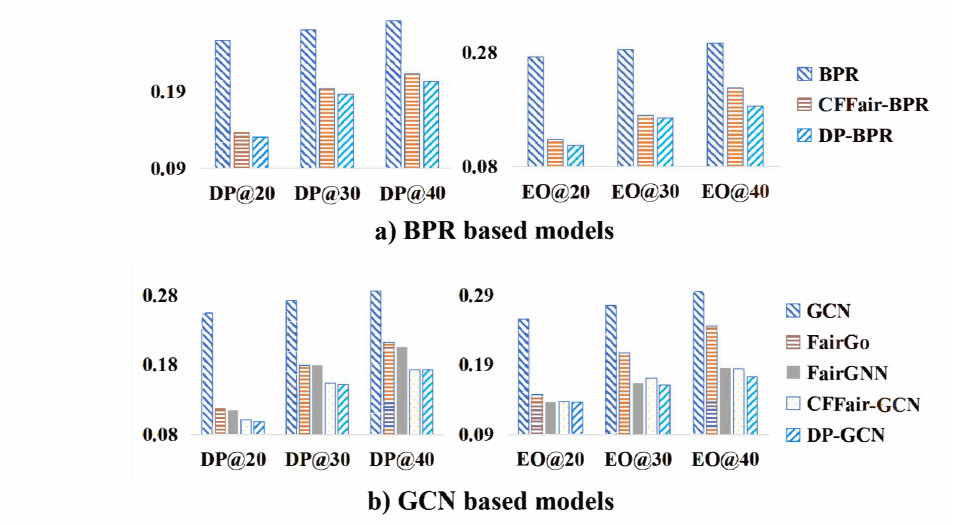


Fig. 4. Statistical fairness performance on the smaller dataset MovieLens-1M with varying K.

Performance on statistical fairness. Statistical fairness metrics have been widely adopted by previous user fairness studies in recommendations [75]. In Table 2 and Table 3, we only compare average counterfactual fairness performance. In order to better display the capability of our proposed *CFFair*, we also evaluate *CFFair* on the statistical fairness metrics on the smaller dataset MovieLens-1M.

A commonly-used metric *Demographic Parity@K (DP@K)* is formulated as

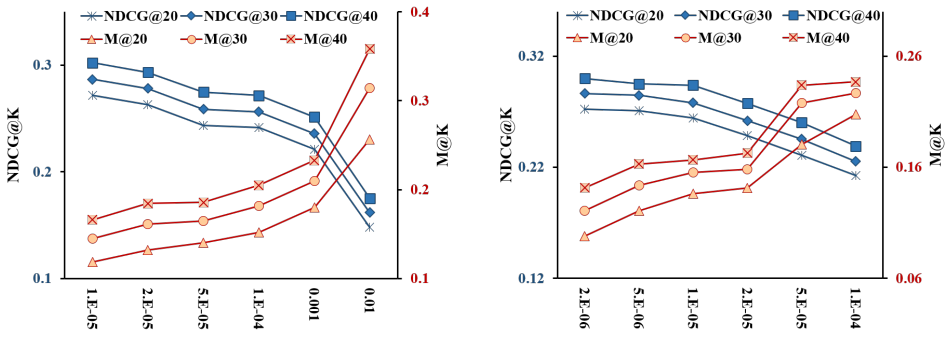
$$DP@K = \frac{1}{N}\sum_{E=1}^{\tilde{\#}} |\mathsf{E}_{D\in^*_1}[1_{E\in\mathcal{L}(D)@}] - \mathsf{E}_{D\in^*_0}[1_{E\in\mathcal{L}(D)@}]|. \tag{27}$$

Specifically, $\mathsf{E}_{D\in^*_1}[1_{E\in\mathcal{L}(D)@}]$ denotes users' average preferences in U_1 for the item v. The quantity can be computed as follows:

$$\mathsf{E}_{D\in^*_1}[1_{E\in\mathcal{L}(D)@}] := \frac{|u:(u,v)\in(u,\mathcal{L}(u)@K)\cap(u\in U_1)|}{|u:(u,v)\in(u,\mathcal{L}(u)@K)|}, \tag{28}$$

Table 8. Performance of different propensity estimators.

Metric	MovieLens-1M					Lastfm-360K				
	LP	MLP	GR	Semi-GCN	*CFFair*	LP	MLP	GR	Semi-GCN	*CFFair*
Accuracy ↑	0.7484	0.7731	0.7923	0.7983	**0.8001**	0.6688	0.6796	0.6922	0.6928	**0.6997**
AUC ↑	0.8133	0.8179	0.8387	0.8392	**0.8415**	0.6924	0.6988	0.7011	0.7041	**0.7156**



(a) *CFFair*-BPR with varying ` (b) *CFFair*-GCN with varying `

Fig. 5. Trade-off effects between recommendation accuracy and average counterfactual fairness on the dataset MovieLens-1M.

where $L(u)@K$ denotes user u's potentially liked list, (i.e., items in the Top-K list for user u), and $|u : (u, v) \in (u, \mathcal{L}(u)@K)|$ denotes the number of users who potentially like item v. Other quantities can be calculated similarly. We modify a statistical fairness metric "value unfairness" [75] in explicit feedback into implicit feedback, and propose *Equal Opportunity@K (EO@K)*, formulated as:

$$EO@K = \frac{1}{N} \sum_{E=1}^{\tilde{\#}} |\mathsf{E}_{D \in {}^*{}_1}[1_{E \in \mathcal{R}_u^{test} \cap \mathcal{L}(D)@} \ ] - \mathsf{E}_{D \in {}^*{}_0}[1_{E \in \mathcal{R}_u^{test} \cap \mathcal{L}(D)@} \ ]|, \tag{29}$$

Note that, the difference between DP@K and EO@K lies in whether focusing on unfairness of predicted results or predicted accuracy. EO@K focuses on the intersection set of Top-K recommended items (i.e., predicted results) and testing items (i.e., the ground truth). Quantities in EO@K (Eq.29) can be calculated similarly as DP@K (Eq.28). $\mathcal{R}_D^{C4BC}$ denotes items in testing data for user u. $v \in \mathcal{R}_D^{C4BC} \cap \mathcal{L}_D@K$ denotes that item v belongs to the intersection of $L_D@K$ and $\mathcal{R}_D^{C4BC}$. It can be seen as "a correct recommendation" between user u and item v. The smaller DP@K and EO@K denote better performance on statistical fairness.

As shown in Figure 4, *CFFair* performs better on statistical fairness metrics than base models, but not as well as statistical fairness-oriented models, i.e., BPR_DP, GCN_DP. These results demonstrate that partial overlap exists between statistical fairness and counterfactual fairness. In other words, optimizing one of the two fairness requirements will also appropriately improve the performance of the other requirement, but performance of the other requirement will decrease.

Parameter sensitivity analysis. The balancing parameter μ in Eq.16 controls the trade-off effects between recommendation performance and average counterfactual fairness. In order to further verify the trade-off effects, we conduct additional experiments on the smaller dataset MovieLens-1M with searching μ in the range of $\{1 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}, 0.001, 0.01\}$ for *CFFair*-BPR, and searching μ in the range of $\{2 \times 10^{-6}, 5 \times 10^{-6}, 1 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}\}$ for *CFFair*-GCN. The corresponding recommendation performance and average counterfactual fairness are recorded in Figure 5.

From Figure 5, we have two observations. First, we can observe obvious trade-off effects between accuracy and fairness performance. As μ increases, both *CFFair*-BPR and *CFFair*-GCN perform better on fairness (M@K increases) but suffer a decrease in accuracy (NDCG@K decreases). Second, *CFFair*-BPR becomes to suffer an obvious accuracy decrease when μ becomes 0.01, and *CFFair*-GCN suffers a decrease when μ is larger than 2×10^{-5}. The results prove that GCN is more fragile to

Table 9. Performance of BPR-based models on matching-based metrics on the PISA-Australia dataset with varying threshold τ. We highlight the best results with bold font.

	Matching ↓				M@20 ↑			
τ	0.5	0.6	0.7	0.8	0.5	0.6	0.7	0.8
BPR	2.9954	1.9044	1.5601	1.3659	0.2035	0.2894	0.3490	0.4272
DP-BPR	0.6942	0.2762	0.1367	0.1152	0.3670	0.5072	0.5526	0.6116
CFFair-BPR	**0.3797**	**0.1843**	**0.1200**	**0.1078**	**0.3739**	**0.5152**	**0.6007**	**0.6906**
	M@30↑				M@40↑			
τ	0.5	0.6	0.7	0.8	0.5	0.6	0.7	0.8
BPR	0.2678	0.3556	0.4186	0.4991	0.3299	0.4263	0.4899	0.5609
DP-BPR	0.4702	0.6146	0.6884	0.7321	0.4981	0.6982	0.7583	0.7850
CFFair-BPR	**0.4717**	**0.6525**	**0.7297**	**0.7938**	**0.5375**	**0.7210**	**0.7782**	**0.8308**

Table 10. Performance of GCN-based models on matching-based metrics on the PISA-Australia dataset with varying threshold τ. We highlight the best results with bold font.

	Matching ↓				M@20↑			
τ	0.5	0.6	0.7	0.8	0.5	0.6	0.7	0.8
GCN	2.5911	0.8936	0.4505	0.3647	0.3061	0.4841	0.5499	0.5996
DP-GCN	1.1317	0.3728	0.2183	0.1852	0.3106	0.5077	0.5824	0.6692
CFFair-GCN	**0.8673**	**0.3178**	**0.1768**	**0.1434**	**0.3211**	**0.5181**	**0.6013**	**0.6848**
	M@30↑				M@40↑			
τ	0.5	0.6	0.7	0.8	0.5	0.6	0.7	0.8
GCN	0.3876	0.5880	0.6438	0.6586	0.4462	0.6573	0.7217	0.7451
DP-GCN	0.3979	0.6234	0.6979	0.7592	0.4578	0.6905	0.7591	0.8090
CFFair-GCN	**0.4206**	**0.6549**	**0.7150**	**0.7741**	**0.4978**	**0.7019**	**0.7690**	**0.8208**

the fairness-aware regularization term, therefore, we should search the best balancing parameters more meticulously for *CFFair*-GCN.

Discussion about matching-based metrics. In order to highlight broad effectiveness rather than intentional selection, we have conducted more experiments on the PISA-Australia dataset by adjusting the similarity threshold τ to 0.5, 0.6, 0.7, and 0.8. Note that, a too high threshold would result in very few and ineffective matches, while a too low threshold would lead to overly lenient matching conditions. The results are recorded in Table 9 and 10. There are several observations from these two tables. First, we find that regardless of how the similarity threshold τ is varied, our proposed *CFFair* consistently achieves a significant improvement of over 2% on all matching-based metrics. This fully demonstrates the stable advantage of our proposed *CFFair* on matching-based metrics. Second, as the threshold τ increases, all models show improvements. The reason is that a higher threshold raises the requirements for matching. Hence, fewer pairs can be matched and the average similarity among these matched users will be higher.

Analyese about the sampling methodology. We would like to explain the rationale behind the sampling methodology, which is based on observations of the practical experiments. During the actual training process, we found that using all the data for fairness-aware regularization term calculation (Eq.14) on the Lastfm-360K dataset exceeded the memory capacity of our GPU. This prompts us to consider which items should be considered in the regularization. We find that a large number of ratings for unpopular long-tailed items do not contribute to the fairness goal. Therefore, we design the sampling methodology. The sampling process consists of two steps. In the first step, we filter out products with click counts below a certain threshold to exclude unpopular items. In

Table 11. Average counterfactual fairness performance on PISA-Australia dataset with/without the sampling methodology.

Model	Sampling	$ATE_{\%}\downarrow$	*Matching* ↓	M@K↑		
				K=20	K=30	K=40
DP-BPR	✓	0.0061 ± 0.0015	0.2762 ± 0.0088	0.5072 ± 0.0233	0.6146 ± 0.0150	0.6982 ± 0.0249
DP-BPR	✗	0.0057 ± 0.0012	0.2479 ± 0.0093	0.5194 ± 0.0185	0.637 ± 0.0124	0.7114 ± 0.0107
CFFair-BPR	✓	0.0022 ± 0.0006	0.1843 ± 0.0124	0.5152 ± 0.0152	0.6525 ± 0.0147	0.7210 ± 0.0182
CFFair-BPR	✗	0.0014 ± 0.0004	0.1665 ± 0.0112	0.5296 ± 0.0169	0.6557 ± 0.0134	0.7246 ± 0.0091
DP-GCN	✓	0.0057 ± 0.0010	0.3728 ± 0.0134	0.5077 ± 0.0144	0.6234 ± 0.0102	0.6905 ± 0.0083
DP-GCN	✗	0.0054 ± 0.0013	0.3606 ± 0.0155	0.5289 ± 0.0070	0.6460 ± 0.0121	0.7018 ± 0.0069
CFFair-GCN	✓	0.0031 ± 0.0012	0.3178 ± 0.0193	0.5181 ± 0.0109	0.6549 ± 0.0081	0.7019 ± 0.0162
CFFair-GCN	✗	0.0024 ± 0.0005	0.2822 ± 0.0164	0.5326 ± 0.0126	0.6611 ± 0.0153	0.7028 ± 0.0174

Table 12. Recommendation performance on PISA-Australia dataset with/without the sampling methodology.

Model	Sampling	HR@K↑			NDCG@K↑		
		K=20	K=30	K=40	K=20	K=30	K=40
DP-BPR	✓	0.3417 ± 0.0051	0.3829 ± 0.0046	0.4223 ± 0.0071	0.2341 ± 0.0077	0.2487 ± 0.0052	0.2630 ± 0.0060
DP-BPR	✗	0.3412 ± 0.0042	0.3823 ± 0.0033	0.4207 ± 0.0062	0.2318 ± 0.0081	0.2477 ± 0.0038	0.2615 ± 0.0044
CFFair-BPR	✓	0.3416 ± 0.0063	0.3843 ± 0.0054	0.4228 ± 0.0088	0.2331 ± 0.0042	0.2492 ± 0.0064	0.2626 ± 0.0051
CFFair-BPR	✗	0.3393 ± 0.0092	0.3821 ± 0.0067	0.4205 ± 0.0076	0.2307 ± 0.0046	0.2487 ± 0.0082	0.2614 ± 0.0040
DP-GCN	✓	0.3422 ± 0.0086	0.3913 ± 0.0077	0.4316 ± 0.0099	0.2397 ± 0.0092	0.2559 ± 0.0080	0.2701 ± 0.0075
DP-GCN	✗	0.3411 ± 0.0061	0.3884 ± 0.0089	0.4264 ± 0.0070	0.2384 ± 0.0075	0.2542 ± 0.0036	0.2688 ± 0.0048
CFFair-GCN	✓	0.3481 ± 0.0069	0.3921 ± 0.0045	0.4338 ± 0.0068	0.2419 ± 0.0060	0.2588 ± 0.0076	0.2729 ± 0.0061
CFFair-GCN	✗	0.3470 ± 0.0094	0.3878 ± 0.0028	0.4258 ± 0.0036	0.2415 ± 0.0080	0.2578 ± 0.0034	0.2712 ± 0.0066

the second step, we measure the known behavioral bias between males and females within these popular items and select a few items with the highest biases as the sampled results.

Additionally, we have conducted comparative experiments on the PISA-Australia dataset by adding and removing the sampling methodology to show its impact. The results are presented in Table 11 and 12. In the PISA-Australia dataset, the sampling methodology selects 50 representative items for computing the regularization term. If we remove the methodology, the calculation will be on all items. We find that removing the sampling module would cause much time cost, but only result in a slight decrease in accuracy and a slight improvement in fairness performance. Therefore, we choose to adopt the sampling methodology.

6 CONCLUSION

In this paper, we argued that most current fairness-aware collaborative filtering models only considered fairness from a statistical perspective. To this end, we started from the Rubin-Neyman potential outcome framework, and proved that minimizing causal effects of the sensitive attribute is equal to achieving average counterfactual user fairness in recommendation. Specifically, we adopted inverse propensity scores to estimate the causal effects, and formulated the causal effects as an additional regularization term. To improve the quality of estimation, we proposed a graph self-supervised propensity estimator to accurately estimate propensities with limited sensitive information. Experimental results on three real-world datasets clearly showed the effectiveness of our proposed *CFFair* on average counterfactual fairness.

ACKNOWLEDGMENTS

This work is supported in part by grants from the National Key Research and Development Program of China (Grant No. 2021ZD0111802), the National Natural Science Foundation of China (Grant No. U23B2031, 7218810119, 62376086, U22A2094).

REFERENCES

[1] Enrique Amigó, Yashar Deldjoo, Stefano Mizzaro, and Alejandro Bellogín. 2023. A unifying and general account of fairness measurement in recommender systems. *Information Processing & Management* 60, 1 (2023), 103115.
[2] Yahav Bechavod, Christopher Jung, and Steven Z Wu. 2020. Metric-Free Individual Fairness in Online Learning. In *NIPS*, Vol. 33.
[3] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research* 7, 11 (2006).
[4] 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 *SIGKDD*. 2212–2220.
[5] Avishek Joey Bose and William Hamilton. 2019. Compositional fairness constraints for graph embeddings. In *ICML*. 715–724.
[6] Jason Brownlee. 2020. *Imbalanced classification with python: Better metrics, balance skewed classes, cost-sensitive learning*. Machine Learning Mastery.
[7] Desheng Cai, Shengsheng Qian, Quan Fang, Jun Hu, and Changsheng Xu. 2022. User Cold-start Recommendation via Inductive Heterogeneous Graph Neural Network. *ACM Transactions on Information Systems* (2022).
[8] Òscar Celma Herrada et al. 2009. *Music recommendation and discovery in the long tail*. Universitat Pompeu Fabra.
[9] 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. *ACM Transactions on Information Systems* (2020).
[10] Lei Chen, Le Wu, Kun Zhang, Richang Hong, Defu Lian, Zhiqiang Zhang, Jun Zhou, and Meng Wang. 2023. Improving recommendation fairness via data augmentation. In *Proceedings of the ACM Web Conference 2023*. 1012–1020.
[11] Wei Chen, Yiqing Wu, Zhao Zhang, Fuzhen Zhuang, Zhongshi He, Ruobing Xie, and Feng xia. 2023. FairGap: Fairness-aware Recommendation via Generating Counterfactual Graph. *ACM Transactions on Information Systems* (2023).
[12] Silvia Chiappa. 2019. Path-specific counterfactual fairness. In *AAAI*. 7801–7808.
[13] Enyan Dai and Suhang Wang. 2021. Say No to the Discrimination: Learning Fair Graph Neural Networks with Limited Sensitive Attribute Information. In *WSDM*. 680–688.
[14] Leander De Schutter and David De Cremer. 2023. How counterfactual fairness modelling in algorithms can promote ethical decision-making. *International Journal of Human–Computer Interaction* (2023), 1–12.
[15] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. 2023. Fairness in recommender systems: research landscape and future directions. *User Modeling and User-Adapted Interaction* (2023), 1–50.
[16] Leyan Deng, Defu Lian, Chenwang Wu, and Enhong Chen. 2022. Graph convolution network based recommender systems: Learning guarantee and item mixture powered strategy. *Advances in Neural Information Processing Systems* 35 (2022), 3900–3912.
[17] Sihao Ding, Fuli Feng, Xiangnan He, Yong Liao, Jun Shi, and Yongdong Zhang. 2022. Causal incremental graph convolution for recommender system retraining. *IEEE Transactions on Neural Networks and Learning Systems* (2022).
[18] Michael D Ekstrand, Anubrata Das, Robin Burke, Fernando Diaz, et al. 2022. Fairness in information access systems. *Foundations and Trends® in Information Retrieval* 16, 1-2 (2022), 1–177.
[19] 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*. 242–250.
[20] Michele Jonsson Funk, Daniel Westreich, Chris Wiesen, Til Stürmer, M Alan Brookhart, and Marie Davidian. 2011. Doubly robust estimation of causal effects. *American journal of epidemiology* 173, 7 (2011), 761–767.
[21] Moritz Hardt, Eric Price, and Nati Srebro. 2016. Equality of opportunity in supervised learning. In *NIPS*. 3315–3323.
[22] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *TIIS* 5, 4 (2015), 1–19.
[23] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*. 639–648.
[24] Xiangnan He, Yang Zhang, Fuli Feng, Chonggang Song, Lingling Yi, Guohui Ling, and Yongdong Zhang. 2022. Addressing Confounding Feature Issue for Causal Recommendation. *ACM Transactions on Information Systems* (2022).
[25] Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press.

[26] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[27] Michal Kosinski, David Stillwell, and Thore Graepel. 2013. Private traits and attributes are predictable from digital records of human behavior. *Proceedings of the national academy of sciences* 110, 15 (2013), 5802–5805.

[28] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences* 116, 10 (2019), 4156–4165.

[29] Matt Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *NIPS*. 4069–4079.

[30] Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. 2019. ifair: Learning individually fair data representations for algorithmic decision making. In *ICDE*. 1334–1345.

[31] Anja Lambrecht and Catherine Tucker. 2019. Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads. *Management science* 65, 7 (2019), 2966–2981.

[32] Chen Lei, Wu Le, Hong Richang, Zhang Kun, and Wang Meng. 2020. Revisiting Graph based Collaborative Filtering: A Linear Residual Graph Convolutional Network Approach. In *AAAI*. 27–34.

[33] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2021. Towards Personalized Fairness Based on Causal Notion. In *SIGIR*. 1054–1063.

[34] Dawen Liang, Laurent Charlin, and David M Blei. 2016. Causal inference for recommendation. In *Workshop at UAI*.

[35] Dugang Liu, Pengxiang Cheng, Zhenhua Dong, Xiuqiang He, Weike Pan, and Zhong Ming. 2020. A general knowledge distillation framework for counterfactual recommendation via uniform data. In *SIGIR*. 831–840.

[36] Mengfan Liu, Pengyang Shao, and Kun Zhang. 2021. Graph-based exercise-and knowledge-aware learning network for student performance prediction. In *Artificial Intelligence: First CAAI International Conference, CICAI 2021, Hangzhou, China, June 5–6, 2021, Proceedings, Part I 1*. Springer, 27–38.

[37] Ou Lydia Liu and Mark Wilson. 2009. Gender differences in large-scale math assessments: PISA trend 2000 and 2003. *Applied Measurement in Education* 22, 2 (2009), 164–184.

[38] Orestis Loukas and Ho-Ryun Chung. 2023. Demographic Parity: Mitigating Biases in Real-World Data. *arXiv preprint arXiv:2309.17347* (2023).

[39] Ting Ma, Longtao Huang, Qianqian Lu, and Songlin Hu. 2022. KR-GCN: Knowledge-aware Reasoning with Graph Convolution Network for Explainable Recommendation. *ACM Transactions on Information Systems* (2022).

[40] Mary Madden, Amanda Lenhart, Sandra Cortesi, Urs Gasser, Maeve Duggan, Aaron Smith, and Meredith Beaton. 2013. Teens, social media, and privacy. *Pew Research Center* 21, 1055 (2013), 2–86.

[41] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. 2018. Learning adversarially fair and transferable representations. In *ICML*. 3381–3390.

[42] Christos Andreas Makridis, Anthony Boese, Rafael Fricks, Don Workman, Molly Klote, Joshua Mueller, Isabel J Hildebrandt, Michael Kim, and Gil Alterovitz. 2023. Informing the ethical review of human subjects research utilizing artificial intelligence. *Frontiers in Computer Science* 5 (2023), 1235226.

[43] Sushmita Mitra and Sankar K Pal. 1995. Fuzzy multi-layer perceptron, inferencing and rule generation. *IEEE Transactions on Neural Networks* 6, 1 (1995), 51–63.

[44] Andriy Mnih and Russ R Salakhutdinov. 2007. Probabilistic matrix factorization. In *NIPS*, Vol. 20.

[45] Debarghya Mukherjee, Mikhail Yurochkin, Moulinath Banerjee, and Yuekai Sun. 2020. Two simple ways to learn individual fairness metrics from data. In *International Conference on Machine Learning*. PMLR, 7097–7107.

[46] Judea Pearl, Madelyn Glymour, and Nicholas P Jewell. 2016. *Causal inference in statistics: A primer*. John Wiley & Sons.

[47] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. 2008. Discrimination-aware data mining. In *SIGKDD*. 560–568.

[48] Felix Petersen, Debarghya Mukherjee, Yuekai Sun, and Mikhail Yurochkin. 2021. Post-processing for individual fairness. *Advances in Neural Information Processing Systems* 34 (2021), 25944–25955.

[49] Christine Pinney, Amifa Raj, Alex Hanna, and Michael D Ekstrand. 2023. Much Ado About Gender: Current Practices and Future Recommendations for Appropriate Gender-Aware Information Access. In *Proceedings of the 2023 Conference on Human Information Interaction and Retrieval*. 269–279.

[50] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. 452–461.

[51] James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. 1994. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American statistical Association* 89, 427 (1994), 846–866.

[52] Lucas Rosenblatt and R Teal Witter. 2023. Counterfactual fairness is basically demographic parity. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 14461–14469.

[53] Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. *J. Amer. Statist. Assoc.* 100, 469 (2005), 322–331.

[54] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *WSDM*. 501–509.

[55] Masahiro Sato, Sho Takemori, Janmajay Singh, and Tomoko Ohkuma. 2020. Unbiased learning for the causal effect of recommendation. In *RecSys*. 378–387.

[56] Jasjeet S Sekhon. 2008. The Neyman-Rubin model of causal inference and estimation via matching methods. *The Oxford handbook of political methodology* 2 (2008), 1–32.

[57] Pengyang Shao, Le Wu, Lei Chen, Kun Zhang, and Meng Wang. 2022. FairCF: fairness-aware collaborative filtering. *Science China Information Sciences* 65, 12 (2022), 1–15.

[58] Jie Shuai, Le Wu, Kun Zhang, Peijie Sun, Richang Hong, and Meng Wang. 2023. Topic-enhanced Graph Neural Networks for Extraction-based Explainable Recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1188–1197.

[59] Latanya Sweeney. 2013. Discrimination in online ad delivery. *Commun. ACM* 56, 5 (2013), 44–54.

[60] Fei Wang, Qi Liu, Enhong Chen, Zhenya Huang, Yuying Chen, Yu Yin, Zai Huang, and Shijin Wang. 2020. Neural cognitive diagnosis for intelligent education systems. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 6153–6161.

[61] Fei Wang and Changshui Zhang. 2006. Label propagation through linear neighborhoods. In *ICML*. 985–992.

[62] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In *SIGIR*. 1288–1297.

[63] Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to Rank with Selection Bias in Personal Search. In *SIGIR*. 115–124.

[64] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *SIGIR*. 165–174.

[65] Yifan Wang, Weizhi Ma, Min Zhang*, Yiqun Liu, and Shaoping Ma. 2022. A survey on the fairness of recommender systems. *ACM Journal of the ACM (JACM)* (2022).

[66] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. 2021. Fairness-aware News Recommendation with Decomposed Adversarial Learning. In *AAAI*. 4462–4469.

[67] Haolun Wu, Chen Ma, Bhaskar Mitra, Fernando Diaz, and Xue Liu. 2022. A Multi-objective Optimization Framework for Multi-stakeholder Fairness-aware Recommendation. *ACM Transactions on Information Systems* 41, 2 (2022), 1–29.

[68] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised graph learning for recommendation. In *SIGIR*. 726–735.

[69] 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*. 2198–2208.

[70] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. 2022. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[71] Yongkai Wu, Lu Zhang, and Xintao Wu. 2019. Counterfactual fairness: Unidentification, bound and algorithm. In *IJCAI*. 1438–1444.

[72] Xin Xin, Jiyuan Yang, Hanbing Wang, Jun Ma, Pengjie Ren, Hengliang Luo, Xinlei Shi, Zhumin Chen, and Zhaochun Ren. 2022. On the User Behavior Leakage from Recommender System Exposure. *ACM Transactions on Information Systems* (2022).

[73] Yonghui Yang, Le Wu, Richang Hong, Kun Zhang, and Meng Wang. 2021. Enhanced Graph Learning for Collaborative Filtering via Mutual Information Maximization. In *SIGIR*. 71–80.

[74] Yonghui Yang, Zhengwei Wu, Le Wu, Kun Zhang, Richang Hong, Zhiqiang Zhang, Jun Zhou, and Meng Wang. 2023. Generative-contrastive graph learning for recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1117–1126.

[75] Sirui Yao and Bert Huang. 2017. Beyond parity: Fairness objectives for collaborative filtering. In *NIPS*. 2921–2930.

[76] Vithya Yogarajan, Gillian Dobbie, Sharon Leitch, Te Taka Keegan, Joshua Bensemann, Michael Witbrock, Varsha Asrani, and David Reith. 2022. Data and model bias in artificial intelligence for healthcare applications in New Zealand. *Frontiers in Computer Science* 4 (2022), 1070493.

[77] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. 2020. Graph contrastive learning with augmentations. *NIPS* 33 (2020), 5812–5823.

[78] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *ICML*. 325–333.

[79] Chen Zhao, Le Wu, Pengyang Shao, Kun Zhang, Richang Hong, and Meng Wang. 2023. Fair representation learning for recommendation: A mutual information perspective. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 4911–4919.

[80] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *SIGKDD*. 1059–1068.