Make Fairness More Fair: Fair Item Utility Estimation and Exposure Re-Distribution

Jiayin Wang
DCST, BNRist, Tsinghua University
Beijing 100084, China
jiayin-w20@mails.tsinghua.edu.cn

Weizhi Ma
AIR, Tsinghua University
Beijing 100084, China
mawz@tsinghua.edu.cn

Jiayu Li
DCST, BNRist, Tsinghua University
Beijing 100084, China
jy-li20@mails.tsinghua.edu.cn

Hongyu Lu
DCST, BNRist, Tsinghua University
Beijing 100084, China
luhy16@mails.tsinghua.edu.cn

Min Zhang*
DCST, BNRist, Tsinghua University
Beijing 100084, China
z-m@tsinghua.edu.cn

Biao Li
Kuaishou Inc.
Beijing 100085, China
biaoli6@139.com

Yiqun Liu
DCST, BNRist, Tsinghua University
Beijing 100084, China
yiqunliu@tsinghua.edu.cn

Peng Jiang
Kuaishou Inc.
Beijing 100085, China
jp2006@139.com

Shaoping Ma
DCST, BNRist, Tsinghua University
Beijing 100084, China
msp@tsinghua.edu.cn

ABSTRACT

The item fairness issue has become one of the significant concerns with the development of recommender systems in recent years, focusing on whether items' exposures are consistent with their utilities. So the measurement of item unfairness depends on the modeling of item utility, and most previous approaches estimated item utility simply based on user-item interaction logs in recommender systems. The Click-through rate (CTR) is the most popular one. However, we argue that these types of item utilities (named *observed utility* here) measurements may result in unfair exposures of items. The number of exposure for each item is uneven, and recommendation methods select the exposure audiences (users).

In this work, we propose the concept of items' *fair utility*, defined as the proportion of users who are interested in the item among all users. Firstly, we conduct a large-scale random exposure experiment to collect the *fair utility* in a real-world recommender application. Significant differences are observed between the *fair utility* and the widely used *observed utility* (CTR). Then, intending to obtain *fair utility* at a low cost, we propose an exploratory task for real-time estimations of *fair utility* with handy historical interaction logs. Encouraging results are achieved, validating the feasibility of *fair utility* projections. Furthermore, we present a fairness-aware re-distribution framework and conduct abundant simulation experiments, adopting *fair utility* to improve fairness and overall recommendation performance at the same time. Online and offline results show that both item fairness and recommendation quality can be improved simultaneously by introducing item *fair utility*.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

Recommender System, Fairness Issues, Item Fairness, Item Utility.

ACM Reference Format:
Jiayin Wang, Weizhi Ma, Jiayu Li, Hongyu Lu, Min Zhang, Biao Li, Yiqun Liu, Peng Jiang, Shaoping Ma. 2022. Make Fairness More Fair: Fair Item Utility Estimation and Exposure Re-Distribution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '22), August 14–18, 2022, Washington, DC, USA*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3534678.3539354

1 INTRODUCTION

Recommender systems have become one of the main approaches for people to acquire information, which utilize the historical interactions between users and items to model user preferences and item representations. In recent years, researchers have found that the exposure and training process of recommender systems suffers from a critical issue, unfairness, for both users and items [14, 27].

Many efforts have been conducted on user unfairness [8, 10, 18, 18, 43], while relatively fewer studies focus on item fairness issues. Existing item fairness studies proposed that the exposure of each item should be positively correlated to its utility to achieve item fairness [7, 25]. The set of widely used item utility measurements is based on historical interaction logs, e.g., click-through rate (CTR), which we name **observed utility** in this paper. However, these studies ignore a large amount of unevenly distributed missing interactions in the user-item space. For example, suppose there are two items, A and B. In history, item A got more exposure data for training, resulting in better performance. So a higher utility can be observed for item A based on historical logs (see Figure 1(a)). However, if two items are exposed to all users, item B will have higher utility than item A (see Figure 1(b)). We argue that the utility calculated based on all users reflects the actual item utility, and therefore the observed utility is inaccurate. Hence, optimizing item

* Min Zhang is the corresponding author.

This work is supported by the Natural Science Foundation of China (Grant No. U21B2026, 62002191) and Tsinghua University Guoqiang Research Institute.

KDD '22, August 14–18, 2022, Washington, DC, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9385-0/22/08.
https://doi.org/10.1145/3534678.3539354

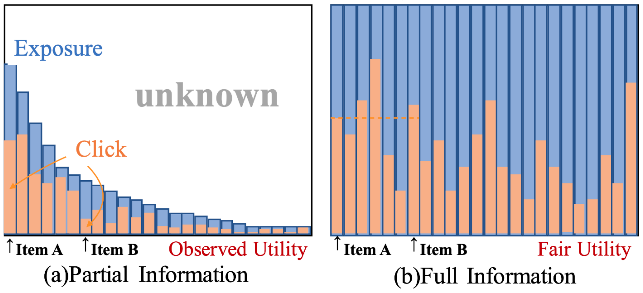


Figure 1: Observed Utility based on partial information vs. Fair Utility based on full information. We cannot obtain the fair utility directly from historical logs, since many interactions are missing, i.e, the white area in (a).

fairness based on observed item utilities is unfair in the first place.

Motivated by the above observations, we propose to model items' **fair utility**, which is defined as the proportion of users who are interested in the item among all users. We believe that an ideal recommender system should give exposures based on this fair utility, i.e., giving item exposure to and only to the users who prefer it. This will achieve both optimal fairness and system performance.

However, it is challenging to estimate the fair utility since it is impossible to show an item to all users and collect their preferences. In terms of probability, we can expose an item to users randomly to approximate its fair utility since the probability of a randomly selected user liking the item is the same as the proportion of the item's candidate users among all users. However, it is costly and raises the concern of hurting the user experience to conduct random exposure experiments. Therefore, it is valuable to estimate such **fair utility** based on the historically **observed utility** sequences.

In this paper, we aim to answer the following research questions:

(RQ1) Is the fair utility consistent with observed utility measurements used in previous works?

(RQ2) Is the item fair utility predictable based on the historical sequence of observed utility?

(RQ3) Is the fair utility helpful to improve the item fairness of recommender systems, as well as the overall system performance?

For the first research question, we clarify the definition of item observed utility and fair utility. Then, we conduct a large-scale randomized experiment to estimate the fair utility in a real-world mobile application for short video recommendations. We investigate the differences between the fair utility obtained from the random experiment and the observed utility from the online system. For the second research question, we model the historically observed utility in a sequence to predict the fair utility. For the third question, we propose a fairness-aware framework that re-distributes system exposures with fair utility for online and offline scenarios. We conduct abundant simulation experiments based on this framework to assess the impact on recommended performance and system fairness.

To summarize, our main contributions are as follows:

(1) To the best of our knowledge, this is the first work to investigate the potential inaccuracy of previous item utility modelings. We refine the utility-based fairness measurements with the item fair utility.

(2) A large-scale random experiment is conducted to estimate item fair utility in an online system and will be released publicly. For low-cost estimations of the fair utility, we experiment with handy observed utility sequences to predict the fair utility and achieve encouraging results.

(3) We present a fairness-aware re-distribution framework with the proposed fair utility. Simulation experiments demonstrate its ability to achieve both item fairness and recommendation performance improvements, which also show that the two metrics are not in conflict but can be improved simultaneously.

2 RELATED WORK

In general, fairness issues are often associated with biases in recommender systems. There are two main purposes for works related to bias and fairness: de-bias for improving recommendation quality or removing biases for fairness concerns. Therefore, we include two subsections of related work, the *Bias and De-biasing* aiming at improving recommendation quality, and the *Fairness* for reducing unfairness in the recommender system.

2.1 Fairness

As recommender systems play an essential role in our lives, it is critical to study and mitigate the unfairness issues involved. Burke [4] classifies fairness in recommender systems as *Consumer Fairness* (also called *User Fairness* [18, 20–22, 39]) and *Provider Fairness* (related to item fairness [2, 11, 13]) based on subjects. There are also works focused on both sides [30], and the integrated fairness between users and items is referred to as *Market Fairness* [34] or *Two-sided Fairness*[27, 40].

From the granularity aspect, the definitions of fairness in recommender systems are usually categorized into individual fairness [9] and group fairness [28, 29, 33] (divided based on sensitive attributes such as user gender, user age, item category, etc. [10, 14, 15, 43, 46]). Both require similar individuals/groups should be treated similarly [9, 28]. The utility measures similarities of quality and treatments are quantified by resources received. For the item side, utilities are mainly measured by CTR or 1 (pursuing equality among all subjects [47]),and resources contain predicted scores,,[14], errors of predicted scores[43] and exposures[34]. Accordingly, the unfairness degree is valued as the disparity of **utility-normalized resources**.

To overcome the problem of unfairness, previous works proposed solutions in various directions: (1) data-based optimization [10, 30], which attempts to modify the distribution of training data to reduce unfairness; (2) model-based optimization, which adjusts the loss function [2, 5, 14, 15, 34, 43] or model structures [46] to improve fairness; (3) outcome-based optimization [12, 16, 27, 32], which reranks the original unfair results to conduct fair recommendation. Counterfactual methods are also often used as one of the technical tools [24, 26, 38].

Our work investigates item fairness among groups categorized by content, using CTR as utility metrics and the number of exposures as resources for items. Unlike previous studies that ignored the gap between the utility obtained from system logs and the actual utility in the whole user space, we refine the measurement of item utility with the Fair Utility. Besides, we propose a fairness-aware re-distribution framework and replace the inaccurate observed utility in fairness-aware methods with our predicted and randomly collected actual fair utility to achieve more precise improvements.

2.2 Relationship with Bias and De-biasing

The observed utilities of items, e.g., CTR, are inaccurate as they may be influenced by exposure/popularity biases or the recommendation methods. Although our study aims to improve item fairness, we want to make clear the differences between our work and these studies.

Based on whether overall unbiased can be achieved, we separate de-biasing works into two types: global strategies, which automatically avoid biases with the calibration of costly and limited experimental uniform data and non-global methods, which focus on specific kinds of known biases [6, 23]. Exposure and Popularity biases are two major known biases related to our fair utility estimation scenario. The former happens as users are only exposed to a part of specific items [31, 37, 41] and the latter refers to the over-recommendation of popular items [44, 45].

In this paper, we focus on item fairness. This purpose is fundamentally different from major de-biasing works, which pursue higher recommendations accuracy and lack fairness considerations. The experiment data we utilized is also different in settings. Instead of exposing randomly in the limited user-item space, we use the feedback of each item randomly exposed to all users in the whole system. The experiment data is very sparse on the user-side, adding negligible impact on each user's experience, and is not suitable for global two-sided de-biasing methods. We also use the latest de-biasing method for popularity bias during training in our experiments. The results indicate that mitigating popularity bias does not necessarily guarantee item fairness improvements.

3 PROBLEM FORMULATION

This section describes the concepts used in this work, including item fairness, observed utility, and fair utility.

3.1 Item Fairness

We follow previous studies [3, 25] that use utility-based fairness as the metric of item fairness, which means each group of items gets resources proportional to its utility. Besides, this metric is also called merit-based fairness[42] or quality-weighted fairness[40].

Formally, we define item fairness problems on groups of items $G = \{G_1, G_2, ...\}$, with the goal that resources received by each group, $R(G) = \{R(G_1), R(G_2), ...\}$, need to be positively correlated with the their utility $U(G) = \{U(G_1), U(G_2), ...\}$. For group $G_x = \{i_{x1}, i_{x2}, ...\}$, we calculate the resource and utility based on the average behavior of items in G_m:

$$R(G_x) = \frac{\sum_{i \in G_x} R(i)}{|G_x|} \quad U(G_x) = \frac{\sum_{i \in G_x} U(i)}{|G_x|} \tag{1}$$

For system-level unfairness evaluation among multi-groups, we use the Herfindahl-Hirschman Index (HHI), a standard measure of market concentration [1] to measure group-level disparities. Note that HHI is a normalized factor in the range of 1/|G| to 1, and lower HHI means fairer performances.

$$unfairness(G) = HHI(\frac{R(G)}{U(G)}) = \sum_{G_i \in G} (\frac{R(G_i)/U(G_i)}{\sum_{G_j \in G} R(G_j)/U(G_j)})^2 \tag{2}$$

As mentioned, fairness is closely dependent on utility (eq2), and we argue that utilities derived from historical interaction data are inconsistent with the situation in the whole user-item space. The number of exposure for each item is uneven, and recommendation methods select the exposure audiences (users). When utility is inaccurate, fairness measurement and optimization could be misleading. Therefore, we discuss the definition of utility in the following subsection.

3.2 Observed Utility and Fair Utility

In this work, we view the recommendation process from the item side. Given an item i, and the complete set of users U, the system predicts users' preference on i, and recommends it to the proper users (U_{obs}). The interaction feedback from these users is collected (F_{obs}) (e.g., click, like, comment rate, etc.) and is used to measure the item's utility. The utility obtained by directly measuring observed historical data is called *observed utility* (OU):

$$\mathrm{OU}(i) = \frac{1}{\sum_{u \in U} \mathbb{I}\left[f_{u,i} \in F_{obs}\right]} \sum_{u \in U} \mathbb{I}\left[f_{u,i} \in F_{obs}\right] utility(f_{u,i}) \tag{3}$$

where $\mathbb{I}$ indicates the indicative function and $utility$ refers to the defined individual utility metric based on the single feedback, such as CTR.

However, the observed feedback F_{obs} cannot reflect the actual item utility as there are usually a large number of missing interactions compared with complete information. Moreover, these missing values are not uniformly random sampled but determined by the previous recommender system. Therefore, the observed utility may not reflect the actual utility.

The ideal and fair measurement of the item utility can be defined as the number (ratio) of the users who like the item if we recommend it to all users. We name it as the *fair utility* (FU). The difference between the observed utility and the fair utility comes from the difference between observed feedback F_{obs} and the ideal full feedback F.

$$\begin{aligned} \mathrm{FU}(i) &= \frac{1}{|U|} \sum_{u \in U} utility(f_{u,i}) \\ &= \frac{1}{\sum_{u \in U} \mathbb{I}\left[f_{u,i} \in F\right]} \sum_{u \in U} \mathbb{I}\left[f_{u,i} \in F\right] utility(f_{u,i}) \end{aligned} \tag{4}$$

Accurate, fair utility of all items will lead to a fair recommendation system. While, it depends on full information, which is difficult to obtain in real-world systems.

The following section will introduce the dataset and compare fair utility and observed utility.

4 EMPIRICAL ANALYSIS OF ITEM UTILITY

4.1 Datasets

We need a large dataset containing abundant random exposures to calculate the fair utility. Existing public datasets either only record system exposure results (Amazon, MovieLens) or are limited in

Table 1: Statistics of the Kwai_Fair dataset.

Kwai_Fair	Items	Users	Interactions
System (R_{obs})	12,579	5,698,826	7,826,461
Experiment (R_{rand})	12,749	10,623,645	12,272,143

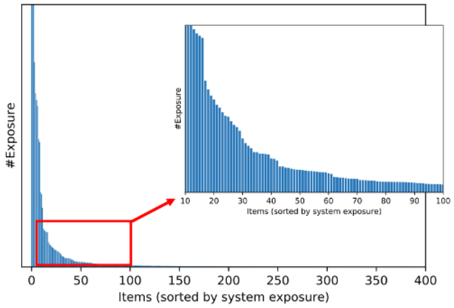


Figure 2: The severely skew distribution of system exposure among items. Long-tailed items receive few resources.

random data size (Yahoo!R3, Coat). So we decide to collect a new dataset for fair utility measurement and counterfactual experiments, namely *Kwai_FAIR* [1]. This dataset, with impressive size (Table 1) and latest information, has high value for related research and will be publicly released along with this paper. It is composed of two parts: regular system interaction logs R_{obs} and stochastic experiment logs R_{rand} of the same group of items, derived from real users of a commercial short video recommendation application on mobile. Each record $r = (u, i, f, t)$ contains the user id, item id, feedback (click, like, comment, watch duration, etc.), and interaction timestamp.

Due to space limitations, detailed information on this Kwai_Fair, including the control of confounding factors in data collection, and more statistics of R_{obs}, R_{rand} are introduced in the appendixA.

To prevent data leakage, we divide items into two groups according to their upload dates. Items uploaded in the first two days construct I_0, and the corresponding data $R_{obs,I_0} = \{r|r.i \in I_0\}$, $R_{rand,I_0} = \{r|r.i \in I_0\}$ is used for the following analysis and training for fair utility estimations. Items uploaded on the third (last) day form I_1. R_{obs,I_1} and R_{rand,I_1} are saved for testing in section 5 and online simulation experiments in section 7.

4.2 Observed Utility vs. Fair Utility (RQ1)

Based on the collected data, we first inspect the exposure distribution of these items, as shown in Figure 2. We can observe that the exposure is significantly skewed. Few items receive most of the recommendation opportunities, while many tail items have barely been exposed. Such a phenomenon indicates concerns on item fairness. *What is the distribution of item utility?*

We then calculate the observed utility and the fair utility. The original system interaction logs measure the observed utility, and experiment data measure the fair utility. The utility of the item is calculated as the click-through rate. The utility distribution shows a different pattern compared to the exposure distribution (shown in Figure 3). High-exposed items do not always have high utility, same as low-exposed items. This confirms the issue of the system's unfairness and optimization space, which is to give opportunities to under-exposed items. Then, does the observed utility match the fair utility?

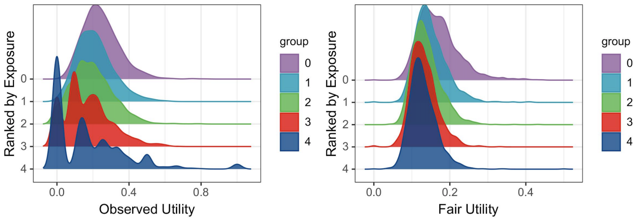


Figure 3: Distribution of Observed utility and Fair utility of items (in terms of Click-Through-Rate). The items are ranked and grouped by the exposure times in the real system in descending order. Exposure is not always aligned with items click utility.

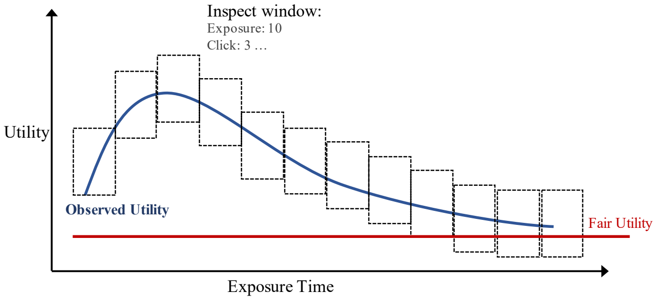


Figure 4: An illustration of fair Utility Estimation: We try to predict the fair utility of the item based on the sequence of observed utility.

We further conduct a comparative analysis and observe significant discrepancies and correlations between the observed and fair utility. Items with higher click probability in system logs are more likely to have a higher random click probability. The correlation between them is confirmed by Pearson's correlation $r = 0.280$. This correlation gives us the insights to estimate the fair utility, which is hard to collect, with the observed utility.

5 EXPLORATORY ESTIMATION OF FAIR UTILITY (RQ2)

The fair utility is necessary for accurate measurements of system fairness. Unluckily, it is impossible to collect it directly in real scenarios because of the high cost of conducting randomized exposure experiments. As shown in the above section, there exists a correlation between the observed utility and fair utility. Therefore, we ask the question: Is the item fair utility predictable based on the historical sequence of observed utility?

5.1 Fair Utility Estimation Task

To start with, we define the fair utility estimation task. Formally, given the sequence of historical observed utility $\vec{OU_i}$ of an item i, we try to estimate its fair utility FU_i

$$\hat{FU_i} = f([OU_{i,1}, OU_{i,2}, ..., OU_{i,i.T}]) \tag{5}$$

Where f indicates models for estimation, $OU_{i,t}$ is observed utility of item i at time t, and $i.T$ is the length of historical observed utility. With the estimation of fair utility $\hat{FU_i}$, we can portray the utility more accurately, which is fundamental for promoting fairness.

[1] The dataset and codes can be found at https://github.com/Alice1998/MakeFairnessMoreFair.

Table 2: The results of fair utility estimation task. The GRU model, which considers the historical observed utility as a sequence performs the best in this task.

(W,S) Models	(10, 10)			(20, 5)			(50, 2)		
	RMSE	MAE	PCC	RMSE	MAE	PCC	RMSE	MAE	PCC
LR	0.0901	0.0718	-0.1117	0.0902	0.0721	-0.1161	0.0903	0.0723	-0.1233
GBDT	0.0891	0.0694	0.2627	0.0801	0.0649	0.4892	0.0798	0.0649	0.5008
GRU	0.0684 (+23.2%)	0.0553 (+20.3%)	0.6414 (+144%)	0.0696 (+13.1%)	0.057 (+12.2%)	0.6468 (+32.2%)	0.0701 (+12.2%)	0.0556 (+14.3%)	0.6085 (+21.5%)

5.2 Methods and Features

We treat this exploratory problem of fair utility estimation as a regression problem, using non-sequential and sequential models for prediction. As our main goal is to verify that the fair utility can be predicted based on the sequence pattern of historically observed utility, but not to propose new models, we use three types of classical regression models for this task:

(1) **Basic model**: linear regression (LR)
(2) **Ensemble model**: Gradient Boosting Decision Tree (GBDT)
(3) **Sequential model**: Gate Recurrent Unit (GRU)

In the sequential model, CTR series are used as input directly. For the basic model and ensemble model, features are concatenated as input. To better understand sequential patterns, we also test 3 different settings for sliding windows, which are listed in Table 2.

5.3 Dataset and Metrics

The task uses both the system and the experiment subsets in XXX_Fair. In the experiment, feedback of items uploaded in the first 2 days (R_{obs,I_0}, R_{rand,I_0}) is used for training and information of the separated items (R_{obs,I_1}, R_{rand,I_1}) composes the test set.

$$F_{obs} = \{r.f \mid r \in R_{obs}\} = \bigcup_{i}^{I} \{f_{i,1},\ f_{i,2}...,\ f_{i,i.T}\}$$

$$OU_{i,t} = utility(f_{i,t}) = f_{i,t}.CTR$$

We consider the mean CTR in F_{rand} for each item as the fair utility FU, which is the target of our estimation. And the sequences of CTR in F_{obs} are valued as the observed utility $\vec{OU}$, which are our inputs. Specifically, given the pre-defined window size W, and step size S, the feedback series are grouped with a sliding window. For each input window $IW_{i,x}$, current CTR and cumulative CTR till this window are considered as two input features. The input window sequences at the length of S compose the final inputs for item i.

$$IW_{i,x} = (\frac{\sum_{t=W*x}^{W*(x+1)-1} OU_{i,t}}{W}, \frac{\sum_{t=0}^{W*(x+1)-1} OU_{i,t}}{W*(x+1)})$$

$$Input_{i,x} = [IW_{i,x}, IW_{i,x+1}, ..., IW_{i,x+S-1}]$$

MSE is used as the loss, and MAE, PCC (Pearson Correlation Coefficient) are used as evaluation metrics.

5.4 Estimation Results

The overall fair utility estimation results are shown in Table 2. The well-performing results conclude that fair utility (FU) can be estimated with the observed utility (OU) series. The sequential model, GRU, which considers changes in the observed utility, performs the best in all metrics and experimental settings, and there is a considerable improvement to non-sequential ones. This result confirms that the sequence pattern of the historically observed utility helps reflect the fair utility. Moreover, the ensemble method GBDT performs better than LR, which indicates that complex non-linear relationships between OU and FU may exist.

As for the three aggregation settings, their inputs are all based on 100 observed utility feedback and the sliding window size are set as 10, 20, and 50, respectively. The test results show that the central setting (with the window size of 20) works the best. On the one hand, adequate records in every window guarantee stability. On the other hand, sliding windows with too large sizes may lose fine-grained information in the changes of observed utility, which leads to a decline in performance. Therefore, we use the 20 group-5 step setting for counterfactual simulation in the following sections.

6 FAIR-AWARE RE-DISTRIBUTION FRAMEWORK (RQ3)

Based on the analysis in Section 4, we find the misordering of item utility and exposure number in the recommender system. This discovery stimulates the win-win possibility for both fairness and system overall recommendation quality (measured by hit rate, CTR, etc.). Accordingly, we design a fairness-aware framework to give the opportunities from over-exposed groups to under-exposed good-quality groups in the re-distribution process.

6.1 Exposure Re-distribution Task

Unlike the re-ranking task that aims to optimize the recommendation results for single users, the target of the exposure re-distribution task is to optimize system-level item fairness and recommendation performance by reordering items.

We first define the re-distribution task as follow: In recommendation systems, original **personalized strategies $\boldsymbol{S_{ori,f}}$** give customized item exposure lists based on user requests. A re-distribution algorithm will use **re-ordering strategies $\boldsymbol{S_{re,g}}$** to reconstruct item exposure sequences $R_{ori,f} = [r_0, r_1, r_2...r_{|R|-1}]$ generated by $S_{ori,f}$ to $R_{re,g} = [r_{g,0}, r_{g,1}...r_{g,|R|-1}]$ for optimizations in item fairness with minimized system quality loss. The main challenge is how to design a good re-distribution method $\boldsymbol{S_{re,g}}$.

In this study, to tackle item unfairness issues, this task re-ranks exposure sequences in group level (divided by item content tag) dynamically based on the information at each current state.

6.2 Re-distribution Strategies

To address the exposure re-distribution task, we further refine our Fairness-aware Re-distribution Framework with trade-off strategies $S_{re,g}$ on item quality and unfairness scores. This framework is designed based on the last ranking stage in real-world online recommendation systems, offering solid possibilities for future migration in online scenarios and offline algorithms.

6.2.1 *Group-level quality and fairness score.*

As our motivation is to mitigate unfairness with minimal negative or even positive impact on recommendation quality, we need to evaluate the status of $R_{re,g}$ dynamically in both recommendation quality and fairness sides. The quality is measured by click-through rate, and the unfairness is represented by an under-exposure degree. For group G at time t,

$$S_{quality,G}(t) = \frac{Click_G(t)}{Exposure_G(t)} \tag{6}$$

$$S_{unfairness,G}(t) = \frac{Utility_G(t)}{Exposure_G(t)} \tag{7}$$

The $Click_G(t)$, $Exposure_G(t)$, and $Utility_G(t)$ are all sum of item-level situations in the group until time t, e.g., $Click_G(t) = \sum_{i \in G} \sum_{t_x \leq t} click_{i,t_x}$. Among them, the $Utility_G(t)$ used in $S_{unfairness,G}$ is a major concern for us. We used the observed utility and predicted fair utility in Section 7 to further validate our estimation models, and used the true fair utility in Section 8 for examination of the regulatory capacities of our fairness-aware framework on different recommendation algorithms $S_{ori,f}$.

6.2.2 *Two-dimensional Modulation.*

To give exposure opportunities to high-quality low-exposure groups, we propose three methods to balance between $S_{quality}$ and $S_{unfairness}$ in the re-distribution experiments. They are listed below:

(1) **Linear**: select the group based on a linear combination of two scores:

$$G_t = \underset{g \in G}{\text{argmax}} \ (\alpha * S_{quality,g}(t) + (1-\alpha) * S_{unfairness,g}(t)) \tag{8}$$

(2) **Quality-ensured**: select the most unfairly treated group in the good-quality candidates

$$G_t = \underset{g \in G}{\text{argmax}}(S_{unfairness,g}(t) * \mathbb{I}[S_{quality,g}(t) > \beta * \overline{S_{quality}(t)}]) \tag{9}$$

(3) **Fairness-ensured**: select the best-quality group in the unfairly treated candidates

$$G_t = \underset{g \in G}{\text{argmax}}(S_{quality,g}(t) * \mathbb{I}[S_{unfairness,g}(t) > \gamma * \overline{S_{unfairness}(t)}]) \tag{10}$$

Where α, β, and γ are hyper-parameters. The framework for our simulation is described in Algorithm 1. To initialize the algorithm, we calculate $S_{quality,0}$ and $S_{unfairness,0}$ with the first K exposures in the original system (Named as *Intervention Time*). And we only consider the first *required* number of exposures. The $Score_g(R_{re,g})$ denotes the score calculated in equation 8, 9, or 10.

6.3 Evaluation Metrics

Since the goal of our exposure re-distribution framework is to optimize fairness with recommendation quality at the same time, we evaluate the re-exposed sequence $R_{re,g}$ with both system quality and fairness situations.

Algorithm 1 Fairness-aware Re-distribution

```
Input: Original exposure sequence $R$, Re-distribution Strategy $S_{re,g}$, Group-level *Utility* (Observed Utility, Fair Utility Prediction Model, true Fair Utility), Intervention Time $K$, Maximal *Required* Exposure Times.
Output: re-distributed exposure sequence $R_{re,g}$ under $S_{re,g}$
initiate $t = K, R_{re,g} = R[R.t \leq K]$
initiate item-level utility and aggregate group-level $Utility$
initiate group-level $S_{quality,0} = Click(K)/Exposure(K)$
initiate group-level $S_{unfairness,0} = Utility(K)/Exposure(K)$
while ($|R_{re,g}| < required$) do
    $t \leftarrow t + 1$.
    select group $\mathcal{G}_{\sqcup} = \text{argmax}(Score_g(R_{re,g}))$.
    select $item_i \in \mathcal{G}_{\sqcup}$ with the highest relevance score (minimal exposure time) in $R - R_{re,g}$, and get the corresponding record $r_{g(t)}$.
    update re-distributed sequence $R_{re,g} = R_{re,g} + [r_{g(t)}]$.
    update $Utility_{\mathcal{G}_{\sqcup}}(t)$, $S_{quality,\mathcal{G}_{\sqcup}}(t)$, $S_{unfairness,\mathcal{G}_{\sqcup}}(t)$.
end while
```

System quality is measured with hit rate and average Click-Through Rate (CTR), and fairness is measured with 1/HHI (introduced in Section 3.1) and 1/MaxDiff. For both metrics, higher values represent better performances (better recommendation quality or fairer in item perspective).

$$r_{unit}(G_i, t) = \frac{R(G_i, t)}{U(G_i, t)} \tag{11}$$

$$\begin{aligned} HHI(t) &= \sum_{G_i \in G} \left(\frac{r_{unit}(G_j, t)}{\sum_{G_j \in G} r_{unit}(G_j, t)}\right)^2 \\ MaxDiff(t) &= \frac{max(r_{unit}(G_x, t)) - min(r_{unit}(G_x, t))}{\sum_{G_j \in G} r_{unit}(G_j, t)} \end{aligned} \tag{12}$$

In the following two sections, we carry out abundant experiments based on our fairness-aware re-distribution framework in both online and offline settings, showing that item fairness and recommendation performance can be improved simultaneously. **Section 7** uses exposure records of the real-world online system as $R_{ori,f}$ and incorporates our fair utility estimation model in $S_{re,g}$ for real-time fairness valuation, validating the feasibility and effectiveness of the re-distribution framework with fair utility. In **Section 8**, we bring state-of-art recommendation algorithms to substitute the online system as $S_{ori,f}$, further validating the generality of our framework on different strategies.

7 ONLINE SIMULATIONS

7.1 Experimental Settings

7.1.1 *Baseline Methods.*

We select five baseline methods for our simulations:

(1) **System**: expose records according to original exposure orders without a group selection process.
(2) **Random**: select groups randomly.
(3) **PD** [44] (latest de-biasing method): leverage popularity bias to remove its bad impact during training.
(4) **Quality-only**: select the group with the top quality.
(5) **Fairness-only**: select the most unfairly treated group.

Table 3: The Fairness-aware Re-distribution is a dynamic regulatory process. We show the snapshot when 20% of system logs are re-exposed and the average performance from beginning to 20% with sample intervals of 1%. System performances are measured by the recommended quality with Hit Rate and CTR, and item-level fairness situations with 1/HHI and 1/MaxDiff (see Eq.12, higher values represent fairer performances). The best-performing method is in bold, and the second-best method is underlined.

$S_{re,g}$	at 20%				Average till 20%			
	HR	CTR	1/HHI	1/MaxD	HR	CTR	1/HHI	1/MaxD
System	0.283	0.263	12.2	4.68	0.278	0.263	11.1	4.69
Random	0.266	0.250	**22.3**	**14.2**	0.265	0.254	22.3	**11.9**
PD	0.323	0.320	7.54	4.39	0.328	0.325	6.61	3.59
Q-only	**0.415**	**0.369**	1.76	1.33	**0.371**	**0.343**	4.10	2.19
F-only	0.281	0.264	21.9	11.4	0.271	0.258	**22.7**	11.7
Linear	0.287	0.269	21.1	9.73	0.279	0.265	22.0	10.3
Q-ensured	0.289	0.270	20.7	10.7	0.278	0.265	21.4	11.1
F-ensured	0.291	0.273	18.8	9.46	0.289	0.276	**22.7**	11.7

Quality-only and Fairness-only can be considered special cases for Linear with $\alpha = 1$ and $\alpha = 0$, respectively. The latest de-biasing method PD is compared in this scenario to demonstrate that eliminating popularity bias does not necessarily enable item-level fairness improvements.

7.1.2 *Parameter Settings.*

In the experiment, we set the intervention time $K = 100$ for each item since the fair utility prediction model requires a minimum of 100 records as input. For evaluations, we snapshot each state when 1% more of re-exposed records is added to $R_{re,g}$. For analysis, we also inspect the average system performances till exposure time at 20%. In pilot experiments, the hyper-parameters α, β and γ in two-dimensional re-distribution strategies $S_{re,g}$ are fine-tuned with grid search in the range of $[0, 1]$, and are set as $\alpha = 0.45, \beta = 0.60, \gamma = 0.65$ for the best model performance. Further parameter-sensitive experiment results are shown in the offline analysis in Section 8.

7.2 Results and Analysis

7.2.1 *Performance Comparison.*

Table 3 summarizes the results of different methods with the predicted fair utility. We have the following observations:

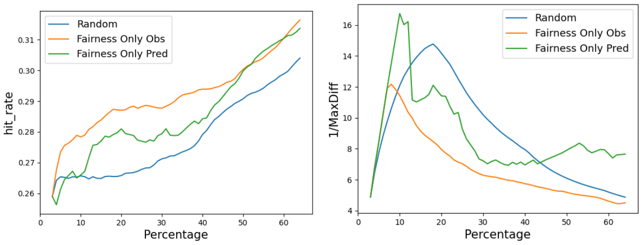
(1) We report both the snapshot when 20% of system logs are re-distributed and the average performance from the beginning till 20% with a sample interval of 1%. In general, most methods achieve steady results across time.

(2) Without incorporating any fairness factors, the Random method and Quality-only method can only optimize one dimension, fairness and recommendation quality, respectively, while hurting another metric significantly.

(3) The de-biasing method (for improving recommendation quality), PD, aims at alleviating popularity bias among items in the training process. It achieves significantly better performance than the original system on recommendation accuracy. However, the item fairness declines significantly. This suggests that adjusting popularity bias does not necessarily guarantee item fairness improvements. Another reason for the poor performance may be that our dataset is sparse on the user side, which does not satisfy the 10-core filtering requirements for each user in this de-biasing method.

Table 4: The best-performing method is in bold and "*" represents a significant improvement in the same strategy using *Obs Utility* or *Pred Fair Utility*. Results illustrate the stable improvement in item fairness by utilizing *Pred Fair Utility*.

Strategy $S_{re,g}$	at 20%		Average till 20%	
	1/HHI	1/MaxDiff	1/HHI	1/MaxDiff
System (wo $S_{re,g}$)	12.2	4.68	11.1	4.69
Fairness-only (Obs)	19.9	7.96	21.8	9.34
Fairness-only (Pred)	**21.9****	**11.4****	**22.7**	**11.7***
Linear (Obs)	19.9	7.96	20.8	8.50
Linear (Pred)	21.1	9.73	22.0	10.3*
Quality-ensured (Obs)	18.8	7.58	20.8	9.07
Quality-ensured (Pred)	20.7**	10.7**	21.4	11.1*
Fairness-ensured (Obs)	15.1	6.08	21.8	9.34
Fairness-ensured (Pred)	18.8**	9.46**	**22.7****	**11.7****



(a) recommendation quality

(b) fairness situation

Figure 5: Comparison between Random and Fairness-only methods with Observed Utility and Predicted Fair Utility.

(4) As for the fairness-aware methods, most of them significantly improve recommendation quality and group-level item fairness compared to the original system. This finding indicates that item fairness and recommendation accuracy can be improved simultaneously. Comparing different methods for incorporating the item fair utility, we find that *Fairness-only* method achieves the most significant fairness improvement and is the only one that compromises the recommendation quality to some extent. On the other hand, *Fairness-ensured* method is the only one achieving significant improvements in both dimensions dynamically.

7.2.2 *Analysis on Different Utility.*

We further inspect how the performances of *Observed Utility* and *Predicted Fair Utility* change. Table 4 presents the comparisons in fairness-aware $S_{re,g}$ and Figure 5 shows the dynamic changes of *Random*, *Fairness-only (Obs)*, and *Fairness-only (Pred)* on recommendation quality and fairness situation. Fairness with *Predicted Fair Utility* quickly surpasses *Observed Utility* as well as random exposure in the early stage, indicating that the measurement of *Predicted Fair Utility* is approaching the true *Item Utility* quickly. Throughout the dynamic process, although the recommendation quality is sightly hurt in the first half process, *Predicted Fair Utility* consistently outperforms the *Fair Utility* in fairness improvements.

Table 5: Performances of fairness-aware re-distribution strategies, $S_{re,g}$ (Linear, Quality-ensured, Fairness-ensured), on three state-of-the-art personalized recommendation methods, $S_{ori,f}$ (BUIR, TiSASRec, KDA), at 20% of system exposure. The best-performing method is in bold, and the second-best method is underlined. The results validate that our framework is effective in improving item fairness on different systems without compromising recommendation quality.

$S_{ori,f}$ / $S_{re,g}$	BUIR				TiSASRec				KDA			
	HR	CTR	1/HHI	1/MaxDiff	HR	CTR	1/HHI	1/MaxDiff	HR	CTR	1/HHI	1/MaxDiff
Base (wo $S_{re,g}$)	0.203	0.126	6.326	4.014	**0.359**	0.223	6.948	3.908	0.372	0.231	7.247	5.138
Random	0.168	0.092	14.48	9.591	0.292	0.184	16.52	15.41	0.340	0.205	13.19	**17.65**
Linear	0.190	0.107	**15.37**	**10.66**	0.320	0.206	**16.94**	**17.14**	0.379	0.226	13.32	16.38
Quality-ensured	**0.246**	**0.176**	7.592	5.498	0.352	0.228	14.33	10.02	**0.396**	**0.249**	11.64	8.225
Fairness-ensured	0.208	0.125	13.41	9.114	0.353	**0.230**	14.19	9.860	0.379	0.226	**13.33**	16.45

These results illustrate that significant and stable improvement in item fairness is achieved with *Predicted Fair Utility*. It further validates the effectiveness of the definition of *Fair Utility* concept and the low-cost Fair Utility prediction method in Section 5.

8 FURTHER OFFLINE ANALYSIS

In this section, we further investigate the usefulness of this framework in a broader range of recommendation systems.

8.1 Exposure Simulation

The real-world online short-video recommender system generates the record sequence R_{obs}, while we hope to simulate more sets of item exposure sequences, $R_{ori,f}$, that different recommendation systems would provide. Based on this goal, we select three lastest algorithms as personalized recommendation strategies $S_{ori,f}$:

(1) **BUIR** [17] (SIGIR 21): latest general algorithm.
(2) **TiSASRec** [19] (WSDM 20): sequential and time-aware recommender method.
(3) **KDA** [35] (TOIS 21): latest time- and knowledge-aware sequential recommendation algorithm.

We only incorporate the item content tag information, which divides items into 15-25 groups, for the knowledge-aware model. Due to space limitations, the detailed $R_{ori,f}$ simulation algorithm under the above three methods is introduced in Appendix B. The output exposure sequence $R_{ori,f}$ contains 18,714 interactions among 9,936 users. Statistics about f's recommendation qualities and the generated item exposure sequences $R_{ori,f}$ are shown in table 6.

Table 6: Information of test results and corresponding exposure sequences under three recommendation algorithms f.

Strategy $S_{ori,f}$	f's Performance in the Test Set				$R_{ori,f}$	
	HR@5	NDCG@5	HR@10	NDCG@10	Item	CTR
BUIR	0.187	0.136	0.218	0.146	128	0.058
TiSASRec	0.377	0.288	0.502	0.329	298	0.124
KDA	0.516	0.371	0.628	0.407	187	0.146

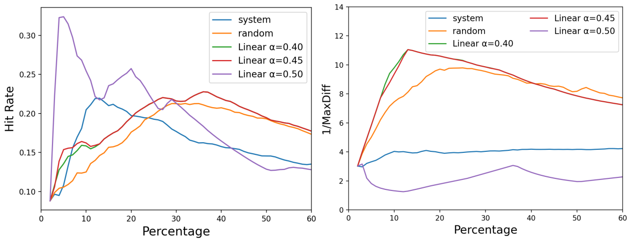
8.2 Experimental Settings

Based on the offline $R_{ori,f}$ of three different recommendation algorithms f, we further conduct simulation experiments with our fairness-aware re-distribution framework. Baseline methods are:

(1) **Base**: direct $S_{ori,f}$ system outputs, without the $S_{re,g}$ process.
(2) **Random**: with the random group selection as $S_{re,g}$.

In the experiments, we set the intervention $k = 3\% * |R_{ori,f}|$, the same as the online simulation in the above section. We also follow the online settings for evaluations, reporting the system (Base) performance at the exposure time of 20%. In pilot studies, hyper-parameters are fine-tuned with grid search in range of [0,1] and are set as $\alpha = 0.45,\ \beta = 0.05,\ \gamma = 0.85$ on all three systems. As we filter the dataset in order to run recommender algorithms f, there is not enough data to support the dynamic fair utility estimation model in the early stages ($|R| = 18,714, |I| = 128 \sim 298$). To avoid data leakage, we split the random experiment data R_{rand} into two subsets for fair utility calculation in simulations and evaluations, respectively.

8.3 Results and Analysis



(a) recommendation quality (b) fairness situation

Figure 6: Grid search for parameter α in method Linear.

Figure 6 shows the parameter-sensitive pilot study for strategy Linear on BUIR. The larger of parameter α, the more $S_{re,g}$ tends to high-quality groups, and conversely, it is more concerned with the under-exposed groups. Accordingly, we observe that the largest setting surpasses the system's recommendation performance while there is no relief on item unfairness. We pick the $\alpha = 0.45$ setting since it achieves significant fairness improvements and performs closest to the system's recommended performance.

Table 5 presents a summary of fairness-aware re-distribution results based on the three different types of state-of-the-art recommendation algorithms. In the horizontal view, the basic model, time-aware sequential model, and time- and knowledge-aware sequential model achieve better recommendation performance one by one both without and with re-distribution strategies. In comparison, this progressive improvement does not hold to fairness performances. In the vertical view, our fairness-aware strategies $S_{re,g}$

are helpful to improve the item fairness, at the same time adding positive or negligible negative impacts on the overall system recommendation performance on all the three personalized $S_{ori,\ell}$ output. These results verify the migration capability of our fairness-aware re-distribution framework as well as the non-conflicting relationships between recommendation performance and item fairness.

9 CONCLUSION

Item fairness is vital for real-world recommender systems, especially for content producers and communities. It is widely agreed that the exposure of each item should be positively correlated to its utility to achieve item fairness, so one of the keys is how to calculate item utility. In this study, we refine the item utility measurement with fair utility, defined as the proportion of users interested in this item among all users. Firstly, a large-scale online random exposure experiment is designed and conducted to estimate fair utility, showing discrepancy and a correlation between observed utility and fair utility. Then we model the handy historical observed utility sequences to the fair utility. At last, we conduct simulation experiments and demonstrate that improvements in item fairness and recommendation quality can be achieved simultaneously based on the predicted fair utility.

REFERENCES

[1] 2018. Herfindahl-Hirschman Index. (2018). https://www.justice.gov/atr/herfindahl-hirschman-index

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

[3] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *SIGIR '18*. 405–414.

[4] Robin Burke. 2017. Multisided fairness for recommendation. (2017).

[5] Robin Burke, Nasim Sonboli, and Aldo Ordonez-Gauger. 2018. Balanced Neighborhoods for Multi-sided Fairness in Recommendation. In *FAT 18 (Proceedings of Machine Learning Research)*, Vol. 81. PMLR, 202–214.

[6] Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. AutoDebias: Learning to debias for recommendation. In *SIGIR 21*. 21–30.

[7] J. Chen, H. Dong, X. Wang, F. Feng, M. Wang, and X. He. 2020. Bias and debias in recommender system: A survey and future directions. (2020).

[8] Yashar Deldjoo, Vito Walter Anelli, Hamed Zamani, Alejandro Bellogín Kouki, and Tommaso Di Noia. 2019. Recommender Systems Fairness Evaluation via Generalized Cross Entropy. In *RecSys 19 (CEUR Workshop Proceedings)*, Vol. 2440.

[9] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. 2012. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. ACM, 214–226.

[10] Michael D. Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D. Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. 2018. All The Cool Kids, How Do They Fit In?: Popularity and Demographic Biases in Recommender Evaluation and Effectiveness. In *FAT 18 (Proceedings of Machine Learning Research)*, Vol. 81. PMLR, 172–186.

[11] 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 18*. ACM, 242–250.

[12] Zuohui Fu, Yikun Xian, Ruoyuan Gao, Jieyu Zhao, Qiaoying Huang, Yingqiang Ge, Shuyuan Xu, Shijie Geng, Chirag Shah, Yongfeng Zhang, and Gerard de Melo. 2020. Fairness-Aware Explainable Recommendation over Knowledge Graphs. In *SIGIR 20*. ACM, 69–78. https://doi.org/10.1145/3397271.3401051

[13] Yingqiang Ge, Shuchang Liu, Ruoyuan Gao, Yikun Xian, Yunqi Li, Xiangyu Zhao, Changhua Pei, Fei Sun, Junfeng Ge, Wenwu Ou, et al. 2021. Towards Long-term Fairness in Recommendation. In *WSDM 21*. 445–453.

[14] Toshihiro Kamishima and Shotaro Akaho. 2017. Considerations on Recommendation Independence for a Find-Good-Items Task. In *Proceedings of Workshop on Responsible Recommendation*. 6. https://doi.org/10.18122/B2871W

[15] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. 2018. Recommendation Independence. In *FAT 18*, Vol. 81. 187–201.

[16] Chen Karako and Putra Manggala. 2018. Using Image Fairness Representations in Diversity-Based Re-ranking for Recommendations. In *UMAP 18*.

[17] D. Lee, S. Kang, H. Ju, C. Park, and H. Yu. 2021. Bootstrapping user and item representations for one-class collaborative filtering. In *SIGIR 21*. 317–326.

[18] Jurek Leonhardt, Avishek Anand, and Megha Khosla. 2018. User Fairness in Recommender Systems. In *WWW 18*. ACM, 101–102.

[19] Jiacheng Li, Yujie Wang, and Julian McAuley. 2020. Time interval aware self-attention for sequential recommendation. In *WSDM 20*. 322–330.

[20] Roger Zhe Li, Julián Urbano, and Alan Hanjalic. 2021. Leave No User Behind: Towards Improving the Utility of Recommender Systems for Non-mainstream Users. In *WSDM 21*. 103–111.

[21] Yunqi Li, Hanxiong Chen, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2021. User-oriented Fairness in Recommendation. In *WWW 21*. 624–632.

[22] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. 2021. Towards Personalized Fairness based on Causal Notion. (2021).

[23] Yunqi Li, Yingqiang Ge, and Yongfeng Zhang. 2021. Tutorial on Fairness of Machine Learning in Recommender Systems. SIGIR.

[24] 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 20*. 831–840.

[25] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. [n. d.]. Controlling Fairness and Bias in Dynamic Learning-to-Rank. In *SIGIR 20*. 10.

[26] Harrie Oosterhuis and Maarten de Rijke. 2021. Unifying online and counterfactual learning to rank: A novel counterfactual estimator that effectively utilizes online interventions. In *KDD*. 463–471.

[27] Gourab K. Patro, Arpita Biswas, Niloy Ganguly, Krishna P. Gummadi, and Abhijnan Chakraborty. 2020. FairRec: Two-Sided Fairness for Personalized Recommendations in Two-Sided Platforms. In *WWW 20*. ACM / IW3C2, 1194–1204.

[28] Dino Pedreschi, Salvatore Ruggieri, and Franco Turini. 2009. Measuring Discrimination in Socially-Sensitive Decision Records. In *SDM 2009*. SIAM, 581–592.

[29] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. 2008. Discrimination-Aware Data Mining. In *KDD 2008 (KDD '08)*. 560–568.

[30] Bashir Rastegarpanah, Krishna P. Gummadi, and Mark Crovella. 2019. Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. In *WSDM 19*. ACM, 231–239.

[31] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *WSDM 20*. 501–509.

[32] Nasim Sonboli, Farzad Eskandanian, Robin Burke, Weiwen Liu, and Bamshad Mobasher. 2020. Opportunistic Multi-aspect Fairness through Personalized Re-ranking. In *UMAP 20*. ACM, 239–247.

[33] Maria Stratigi, Jyrki Nummenmaa, Evaggelia Pitoura, and Kostas Stefanidis. 2020. Fair sequential group recommendations. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*. 1443–1452.

[34] Mengting Wan, Jianmo Ni, Rishabh Misra, and Julian McAuley. 2020. Addressing marketing bias in product recommendations. In *WSDM 20*. 618–626.

[35] Chenyang Wang, Weizhi Ma, Min Zhang, Chong Chen, Yiqun Liu, and Shaoping Ma. 2020. Toward dynamic user intention: Temporal evolutionary effects of item relations in sequential recommendation. *TOIS 20* 39, 2 (2020), 1–33.

[36] Chenyang Wang, Yi Ren, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. 2021. ReChorus: A comprehensive, efficient, flexible lightweight recommendation algorithm framework. *Software Journal* 33, 4 (2021), 0–0.

[37] Xiang Wang, Yaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, and Tat-Seng Chua. 2020. Reinforced negative sampling over knowledge graph for recommendation. In *WWW 20*. 99–109.

[38] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *KDD 21*. 1791–1800.

[39] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. 2021. Fairness-aware News Recommendation with Decomposed Adversarial Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4462–4469.

[40] Y. Wu, J. Cao, G. Xu, and Y. Tan. 2021. TFROM: A Two-sided Fairness-Aware Recommendation Model for Both Customers and Providers. (2021).

[41] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *RecSys 2018*. 279–287.

[42] Tao Yang and Qingyao Ai. 2021. Maximizing Marginal Fairness for Dynamic Learning to Rank. In *WWW 21*. 137–145.

[43] Sirui Yao and Bert Huang. 2017. Beyond Parity: Fairness Objectives for Collaborative Filtering. In *NIPS 17*. 2921–2930.

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

[45] Ziwei Zhu, Yun He, Xing Zhao, Yin Zhang, Jianling Wang, and James Caverlee. 2021. Popularity-Opportunity Bias in Collaborative Filtering. In *WSDM 21*. 85–93.

[46] Ziwei Zhu, Xia Hu, and James Caverlee. 2018. Fairness-Aware Tensor-Based Recommendation. In *CIKM 18*. ACM, 1153–1162.

[47] Ziwei Zhu, Jingu Kim, Trung Nguyen, Aish Fenton, and James Caverlee. 2021. Fairness among New Items in Cold Start Recommender Systems. (2021).

A DATASET

This dataset is collected from a commercial mobile application for short video recommendations. In this online platform, the average length of videos is around 30 seconds, and videos are displayed full-screen.

A.1 Collection

Note that this dataset only records the desensitized user id, item id, and user-allowed interaction information. All the data collection process meets the requirements of relevant laws and regulations.

A.1.1 System Data, R_{obs}.

As warm items have already been recommended to some users, we choose cold items. Every hour, we randomly select 200 new uploaded items and this sample period lasts for three days. For each item, we record sensitive features, including video length, upload time, content tag (games, news, foods, parenting, etc.), author information (number of followers), and collect the regular recommender system interaction records $r = (u, i, f, t)$ within three days from uploading, with user id, item id, feedback, and interaction timestamp.

A.1.2 Experiment Data, R_{rand}.

For uniform feedback, we conduct random exposure experiments. In the online experiments, we control several confounding factors (item uploaded time and exposed time). Specifically, we expose the same group of items in R_{obs} (which are randomly selected) to stochastic users, with the intensity of 20 exposure per hour per item for two days. This experimental flow is randomly mixed into the normal personalized recommendation flow. Therefore, the feedback in R_{rand} reflects users' preferences under natural browsing behavior. The obtained data are first sampled uniformly from the item space and then randomly from the user space. As the probability of relevance for a random user is equal to the relevance ratio in the whole user space, these statistics can approximate complete information.

A.2 Statistics

Table 7: Statistics of item exposure and utility based on observed system and random experiment records after 100-exposure-filter.

	System			Experiment		
	Exp.	**Click**	**Like**	**Exp.**	**Click**	**Like**
mean	4,440	0.2483	0.0281	970	0.1445	0.0040
std	40,592	0.1016	0.0359	230	0.0434	0.0023
min	100	0.0092	0.0000	101	0.0434	0.0023
25%	132	0.1795	0.0057	830	0.1141	0.0024
50%	205	0.2368	0.0152	892	0.1370	0.0037
75%	549	0.3062	0.0370	1,081	0.1667	0.0054
max	1,143,357	0.8357	0.2823	1,621	0.4362	0.0268

The statistics of R_{obs}, R_{rand} are shown in Table 7. Note that a constant is subtracted in the online metrics shown due to commercial consideration, but it does not affect the findings.

B OFFLINE EXPOSURE GENERATION

In this section, we introduce the offline item exposure sequence $R^* = R_{ori,\mathscr{f}}$ generation process under the recommendation system $\mathscr{f}$.

In order to run personalized strategies $S_{ori,\mathscr{f}}$ under leave-one-out strategies, we filter out users with less than three positive or one negative feedback in R_{obs} in our xxx_Fair dataset to get the corresponding subset R'_{obs} with user set U' and item set I'.

Formally, we separate R'_{obs} into the positive interaction list $R'_{obs,P}$ and negative record list $R'_{obs,N}$:

$$R'_{obs,\ P} = R'_{obs}[R'_{obs}.f = \texttt{f.1}]$$
$$R'_{obs,\ N} = R'_{obs}[R'_{obs}.f = \texttt{f.0}]$$

Following the experimental settings in [35, 36], we choose the last interaction in $R'_{obs,\ P}$ for each user as the candidate items in the test set ($R'_{obs,\ test}$), the second last items for each user as candidates in validation set ($R'_{obs,\ valid}$), and the remaining positive interactions in $R'_{obs,\ P}$ as $R'_{obs,\ train}$, the training set. A recommendation system, $\mathscr{f}$, trains its model on $R'_{obs,\ train} \cup R'_{obs,\ valid}$. We recommend top-k (k=5,10,$|I'|$) items from the whole item set I' for each test or validation case. Given the same generated user request list $R'_{obs,req}$, we produce offline exposure sequences $R_{ori,\mathscr{f}}$ under different personalized strategies $S_{ori,\mathscr{f}}$. The whole simulation process is described in Algorithm B.

Algorithm 2 exposure sequence generation

```
Input: training data $R'_{obs,\ train}$, validation data $R'_{obs,\ valid}$, testing
    data $R'_{obs,\ test}$, negative data $R'_{obs,\ N}$, Recommendation System
    (Algorithm) $\mathscr{f}$
Output: offline exposure sequence $R^* = R_{ori,\mathscr{f}}$
generate the user request list:
    $R'_{obs,req} = \{(r.u,\ r.t) | \forall r \in (R'_{obs,\ N} \cup R'_{obs,\ test})\}$
    $R'_{obs,req}.t^* = R'_{obs,req}.groupby(u).t.rank()$
train the recommendation system $\mathscr{f}$ with data $R'_{obs,\ train}$ and
    $R'_{obs,\ valid}$.
run the trained model $\mathscr{f}$ on the test set, with $R'_{obs,\ test}$ as
    ground-truth, all items $I'$ as ranking candidates.
generate the recommended item candidate list $I_{\mathscr{f}}$ for each test
    case (user) $(r.u, r.t)$ and organize the corresponding feedback
    list $F_{\mathscr{f}}$.
produce the exposure sequence based on user request $R'_{obs,req}$
    and personlized item candidate list $I_{\mathscr{f}}$:
$R^* = \{(r.u,\ I_{\mathscr{f}}(r.u)[r.t^*]),\ F_{\mathscr{f}}(r.u)[r.t^*],\ r.t) | \forall r \in R'_{obs,req}\}$
```