# A Novel Shadow Variable Catcher for Addressing Selection Bias in Recommendation Systems

Qingfeng Chen[a], Boquan Wei[a], Debo Cheng[b], Jiuyong Li[b], Lin Liu[b] and Shichao Zhang[c]
[a]School of Computer, Electronics and Information, Guangxi University, Nanning, China
[b]UniSA STEM, University of South Australia, Adelaide, Australia
[c]Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin, China

***Abstract*—Recommender systems rely on observational data to predict user ratings for unseen items. Since the observational data is typically missing not at random (MNAR), they contain biases, predominantly selection bias, thus models trained on such data are inherently biased. If a shadow variable, which is a variable used instead of user's latent variables that influence both the treatment and the outcome, can be identified, it is possible to build unbiased models for recommender systems. To overcome the challenge of manually identifying valid shadow variables in the data, we propose a Shadow Variable Catcher (SVC), a model designed to learn the latent representation of shadow variables from observational data. By using the learned representation of shadow variables by SVC, we develop the Shadow Debiased Recommender (SDR) method to build an unbiased collaborative filtering model for addressing selection bias in recommender systems. Comprehensive experiments on both synthetic and real-world datasets, have verified the performance of SDR and demonstrated its effectiveness and robustness, and offer new insights into the mitigation of bias in recommender systems.**

***Index Terms*—Causal Inference, Recommender Systems, Selection Bias, Shadow Variables**

## I. Introduction

Recommender systems have become indispensable in today's Internet landscape, which significantly enhance user experience by accurately identifying the required information amid a complex network of data [1]–[6]. They are widely used in various domains, including e-commerce [7], social media [3], music [8] and video streaming [9]. In the recommendation domain [10], collaborative filtering models suggest items to users that align with their preferences by analyzing user characteristics, historical interactions, and other relevant data. From the perspective of causal inference, collaborative filtering seeks to estimate the causal effect of system exposure on user feedback. This involves answering the counterfactual question: What kind of feedback would be given if the user were recommended that unseen item? By answering this counterfactual question, we can better understand and improve the effectiveness of recommender systems.

Understanding and addressing the issue of data being missing not at random (MNAR) is crucial for accurately modeling and mitigating selection bias in recommender systems. Ideally, the data collected for recommender systems would be missing at random (MAR), allowing the model to learn real causal effects directly from historical interactions. However, in the real-world, the recommendation data used for training is usually MNAR. In a recommendation data, users tend to rate items they like and are more likely to rate items that are good or bad [5]. These specific missing mechanisms create the collider bias between exposure and ratings, also known as selection bias [10], [11]. This selection bias causes the model to fail in learning the correct rating distribution, especially if the discrepancy between the training rating distribution and the true rating distribution is significant [12], [13]. For example, in Figure 1, we illustrate the difference in the distribution of ratings under random collection and user selection in the Yahoo!R3 dataset. If the missing mechanism can be correctly captured, selection bias can be effectively addressed [14]. Existing research has frequently overlooked that the training data are MNAR and therefore ignores the effects of selection bias. As a result, these algorithms have been adversely affected by selection bias.

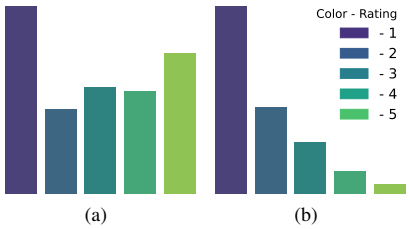


Fig. 1. Distributional differences in ratings between (a) user-selected and (b) randomly collected data in the Yahoo!R3 dataset.

Shadow variables [15] are a special type of covariate characterized by their independence from the missing mechanisms when conditioned on the remaining covariates and outcomes [16]–[18]. Recently, research on leveraging shadow variables to ensure identifiability of models built from MNAR data has gained popularity [19], [20]. Figure 2 shows a simple example for illustrating shadow variables in recommender systems. It is known that user rating preference may cause selection bias, e.g., users tend to prefer rating items they particularly like or particularly dislike [21]. A user's rating preference does not affect the system exposure or the user's rating value, but it constitutes a collider structure, as shown in the causal graph in Figure 2. In this case, user interests is a valid shadow variable. It affects the probability of item exposure and also impacts

*Corresponding authors: Q. Chen (qingfeng@gxu.edu.cn) and D. Cheng (Debo.Cheng@unisa.edu.au).

2024 IEEE International Conference on Data Mining (ICDM) | 979-8-3315-0668-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICDM59182.2024.00014



2374-8486/24/$31.00 ©2024 IEEE
DOI 10.1109/ICDM59182.2024.00014

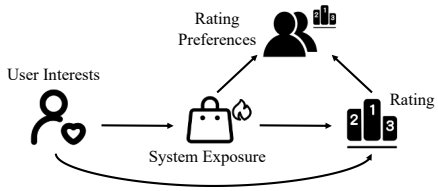


Fig. 2. An example of a shadow variable represented in a causal graph, where Rating Preferences is a collider.

ratings, but not directly related to the user's rating preferences.

To address selection bias in recommender systems, we propose a novel multi-loss model, i.e., Shadow Variable Catcher (SVC), to learn the latent representation of shadow variables from the observational data by using deep generative paradigm [6], [22], [23]. Furthermore, we propose a novel framework, i.e., Shadow Debiased Recommendation (SDR for short), to use the learned representation of shadow variables for addressing the selection bias in recommender systems. Subsequently, we implement unbiased training of the recommendation model based on the shadow variables to obtain unbiased estimator under MNAR data. Our main contributions are listed below:

- We propose the SVC model that captures the latent representation of shadow variables from observational data. The captured shadow variables are highly valuable as a special kind of covariate, as well as proxy variables of unmeasured confounders, for both deconfounding and de-selection bias in recommender systems.
- We propose a novel recommendation framework, SDR, which uses the learned representation of shadow variables to mitigate selection bias in collaborative filtering models. To the best of our knowledge, our SDR model is the first attempt at addressing selection bias using shadow variables in recommender system.
- We conduct extensive experiments on one synthetic and three real-world datasets to validate the performance of the proposed method for debiasing.

## II. RELATED WORK

In this section, we briefly overview the development of causal inference in recommender systems and review related work that addresses selection bias.

### A. Debiasing methods in recommendations

For recommender systems, performance is often affected by various biases due to the difficulty of capturing true relations in biased data. The best way to deal with this problem is to perform randomized experiments such as online A/B testing to collect unbiased data. However, recommendation models are time-sensitive. Running randomized experiments at regular intervals would severely degrade the user experience and impact the system's operation, which is unacceptable. Causal inference [11], [24], a discipline that infers causality from data, has achieved remarkable advances in recent years when used for dealing with bias in recommendations.

There are two fundamental biases [5]: confounding bias caused by unmeasured confounders and selection bias caused by controlled colliders. Leveraging the wealth of research on confounding bias in causal inference, relevant works have proposed numerous methods for addressing confounding bias in recommender systems. The backdoor path-based approaches [25], [26] implement backdoor adjustment to block indirect causal paths. Some methods assume the availability of certain variables within the data, such as instrumental variables (e.g., search logs) or mediators (e.g., click feedback), subsequently performing classic IV-estimation [27], [28] or front door adjustment [29]. Methods learn the representation of unmeasured confounders [30], [31] from proxy variables through generative models. These generative debiasing methods, while capturing the variables implicit in the data to mitigate confounding bias, did not extend the theory and structure further. The study by [32] adopted invariant learning methods to learn users' invariant interests in the presence of unmeasured confounders. The multi-task residual model designed in study [33] also provides a viable solution to unmeasured confounders. In contrast, methods for dealing with selection bias are relatively few and usually require stricter assumptions.

### B. Methods addressing selection bias in recommendations

Selection bias is prevalent in recommender systems where ratings collected in non-random user interactions inevitably introduce selection bias [21]. The propensity-based method IPS [34] uses inverse probability weighting to adjust the model's training loss, providing an unbiased estimation of the true risk. But, propensity-based methods only achieve unbiased estimation if they have the true propensity. As propensity estimation suffering from a high variance problem [35], model performance is not stable. Subsequent work [36] has further improved propensity estimation based on sensitivity analyses, and the work in [37] also proposed a new policy learning to enhance the propensity model. A study [7] decomposed the direct effect of users on ratings as a way to mitigate the impact of user selection issues on the model.

One aspect of the study focuses on the non-random characteristics of missing data, generating pseudo-labels for missing data by an imputation model and then training a recommendation model on a complete matrix, e.g., EIB [13], making the matrix close to the ideal uniform matrix by imputation pseudo-labels. The research in [35] proposed the asymmetric training framework to improve the imputation method. However, the imputation model suffers from empirical inaccuracy and cannot guarantee the availability of an unbiased prediction model. Doubly Robust Joint Learning (DR) [38] combines the benefits of IPS and EIB, allowing unbiased models to be trained as long as the imputed errors or propensities are accurate. Stable-DR [39] provides a more stable DR based on cyclical learning. Different from prior paradigms dealing with selection bias, this paper presents a viable approach to mitigate selection bias from a novel perspective.

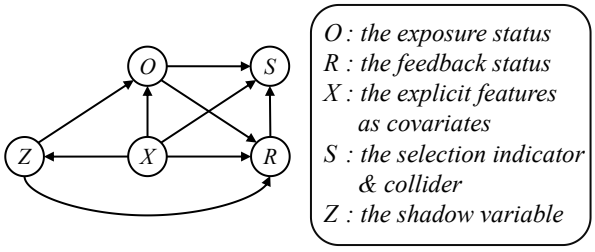


Fig. 3. A causal graph representing the recommendation process, illustrating the relationships between the exposure status ($O$), feedback status ($R$), explicit features as covariate ($X$), selection indicator and collider ($S$), and shadow variable ($Z$).

## III. PROBLEM FORMULATION

In this section, we introduce the basic notations used in our work and formulate recommendation tasks based on causal graphs. We also use causal graphs [11] to describe shadow variables and address the issues caused by selection bias.

### A. Notations

Let $\mathcal{U} = \{u\}$ and $\mathcal{I} = \{i\}$ denote the sets of users and items, respectively. As illustrated in Figure 3, recommendation focuses on the following elements:

- *Unit*: The basic unit is user-item pair $(u, i)$.
- *Covariate*: The explicit features of users and items, denoted as $X = \{X_u, X_i\}$, where $X_u$ represents features of the users and $X_i$ represents features of the items.
- *Treatment*: $O = \{o_{u,i}\}$ is the exposure status. $o_{u,i} = 1$ when the rating of this unit $(u, i)$ is observed, otherwise $o_{u,i} = 0$.
- *Outcome*: $R = \{r_{u,i}\}$ is the feedback status. $r_{u,i}$ represents the feedback rating of user $u$ for item $i$. The ratings $r_{u,i}$ sorted from highest to lowest, form the list of recommendations.
- *Selection Indicator & Collider*: $S = \{s_{k,u,i} \mid k = 1, 2, \cdots, K\}$, where $s_{k,u,i} \in \{0, 1\}, \forall k \in K$, is selection indicator or collider as shown in Figure 3. When the data is MNAR caused by $s_{k,u,i}$, it means that only or most of the data has $s_{k,u,i} = 1$, while for $s_{k,u,i} = 0$, the data is completely missing or minimal. We use $S = 1$ to indicate the presence of a majority of colliders as 1 in the data, i.e., the data is MNAR. Conversely, $S = 0$ denotes the unobserved case where colliders as 0.

Let $P$ denote the probability mass function for discrete variables and $f$ denote the probability density function for continuous variables. Under the recommendation task, our goal is to train a model to predict the user's rating of an item, represented by the probability mass function $P(R = 1 \mid O, X, S)$. A higher rating means that the user is more likely to click or buy the item, thereby creating a Top-K recommendation list. Due to the problem of MNAR data, after training we will only get $P(R = 1 \mid O, X, S = 1)$, implying that some propensity interferes with the causal effect of exposure $O$ on rating $R$.

To eliminate the bias caused by MNAR data, we need to impute the missing data $S = 0$ to adjust the model training. Specifically, we calculate the $S$ average effect based on the conditional probability formula as follows:

$$P(R = 1 \mid O, X, S) = \sum_{d}^{0,1} P(R = 1 \mid O, X, S = d) P(S = d \mid O, X) \tag{1}$$

The main missing component of Eq. (1) is $P(R = 1 \mid O, X, S = 0)$. Solving the MNAR problem is equivalent to recovering $P(R = 1 \mid O, X, S = 0)$ from data.

### B. Shadow variables

The shadow variable is a special type of covariate, e.g., the variable $Z$ represented in the causal graph in Figure 3, which must satisfy three conditional independence assumptions outlined in Assumption 1.

**Assumption 1:** A variable $Z$ is a shadow variable if it satisfies (1) $Z \not\perp\!\!\!\perp O \mid X, R, S = 1$; (2) $Z \not\perp\!\!\!\perp R \mid X, O, S = 1$; and (3) $Z \perp\!\!\!\perp S \mid X, O, R$ [18].

where, $\perp\!\!\!\perp$ denotes independent and $\not\perp\!\!\!\perp$ denotes dependent. The first two conditional independence assumptions represent the relationships of the shadow variable $Z$ with *treament* $O$ and *outcome* $R$, respectively. From a causal graph perspective, shadow variable $Z$ maintains paths to $O$ and $R$ after the other indirect causal paths have been blocked. The last assumption requires that the shadow variable $Z$ has no effect on the collider $S$ when conditioning on the other observed variables. There is no direct causal path from $Z$ to $S$ in the causal graph, they are conditionally independent.

Previous studies [40] have indicated that unbiased models under MNAR data cannot be directly identified without further hypotheses and prior knowledge. However, it is possible to identify true effects if there are shadow variables in the data that correspond to the selection indicator $S$ [19]. Finding convincing shadow variables in datasets collected under the usual operation of recommender systems is challenging. This does not imply that there are no implicit shadow variables lurking behind the observational data. In this paper, we will recover the latent representation of the shadow variables and apply them to calculate the average causal effect, thereby achieving unbiased training with MNAR data.

## IV. THE PROPOSED SDR METHOD

In this section, we first introduce our proposed shadow variable catcher for learning the latent representation of the shadow variables. Then, we describe the method for addressing selection bias with shadow variables. Finally, we introduce our proposed shadow-debiased recommendation by training recommendation model using shadow variables.

### A. Shadow variable catcher (SVC)

In this work, we aim to recover the latent representation of shadow variables via the generative models [22], [41]. We propose a novel generative model (SVC), to learn the latent representation of shadow variables from data directly.

In our SVC model, we employ the Variational Autoencoder (VAE) [22] as our generative model to learn and generate the latent representation of the shadow variables. Note that we

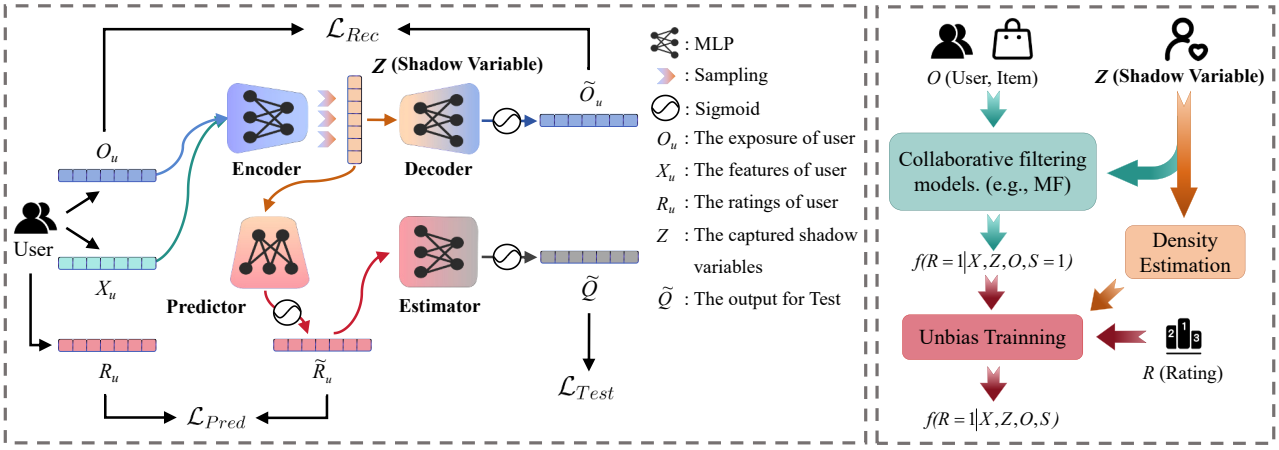


Fig. 4. An overview graph of our proposed SDR model. **Left** graph: The architecture of Shadow Variable Catcher (SVC). **Right** graph: The Processes of Shadow Debiased Recommendation (SDR).

impose the three conditions outlined in Assumption 1 on the measured data to learn shadow variable representation for each user that satisfy these conditions as robustly as possible.

SVC takes user exposure $O_u$ and user features $X_u$ as inputs. The *Encoder* module in our SVC outputs the mean and variance, followed by a reparameterization trick to sample the generative variable. To ensure that there is sufficient information to learn the latent representation $Z$ of shadow variables, we set the prior distribution of the latent representation, which prevents overfitting features in the unordered latent space. Referring to the encoder of the iVAE [42], user features $X_u$ serves as an auxiliary variable in the *Encoder*, providing a source for the prior distribution of the latent variables, thereby ensuring the identifiability of the latent representation $Z$. That is, the prior distribution of $Z$ is modeled as $P_\theta(Z \mid X_u)$.

We assume that the prior distribution $P_\theta(Z \mid X_u)$ belongs to the Gaussian location-scale family. The distribution $q_\phi(Z \mid X_u, O_u)$ is sampled from the approximate posterior. We constrain this posterior distribution $q_\phi(Z \mid X_u, O_u)$ with Kullback-Leibler divergence of two Gaussian distributions:

$$\begin{aligned}\mathcal{L}_{KL} &= -KL(q_\phi(Z \mid X_u, O_u) \| P_\theta(Z \mid X_u)) = \\ &- KL(N(\mu_q(X_u, O_u), \sigma_q^2(X_u, O_u)) \| N(\mu_p(X_u), \sigma_p^2(X_u)))\end{aligned} \tag{2}$$

where $\mu_q$, $\sigma_q^2$, $\mu_p$, and $\sigma_p^2$ are parameters modeled by four different Multilayer Perceptron (MLP) models. Next, we design three constraint terms corresponding to the three conditions in Assumption 1 to generate the latent representation $Z$.

**Loss function $\mathcal{L}_{Rec}$ for condition 1:** $Z \not\perp\!\!\!\perp O \mid X, R, S = 1$**.** The goal of condition 1 is to ensure that $Z$ has enough influence over $O$. Since the latent variables are extracted from the exposure $O$, condition 1 indicates whether $Z$ has sufficient capacity to reconstruct $O$. SVC models the *Decoder* using an MLP and implements this constraint based on binary cross entropy (BCE) loss. The *Decoder* outputs an approximation of $O_u$, denoted as $\tilde{O}_u$. The reconstruction loss can be expressed as $\mathcal{L}_{Rec} = BCE(O_u, \tilde{O}_u)$.

**Loss function $\mathcal{L}_{Pred}$ for condition 2:** $Z \not\perp\!\!\!\perp R \mid X, O, S = 1$**.** Condition 2 aims to verify the effect of $Z$ on the outcome $R$. This is equivalent to the predictive ability of $Z$ for $R$. It is consistent with our final purpose for recommendation task, and therefore it is the primary loss. We adopt an MLP as the *Predictor* of the outcome $R$. The output of the predictor is $\tilde{R}_u$, then, the prediction loss based on BCE loss is $\mathcal{L}_{Pred} = BCE(R_u, \tilde{R}_u)$.

**Loss function $\mathcal{L}_{Test}$ for condition 3:** $Z \perp\!\!\!\perp S \mid X, O, R$**.** Condition 3 requires that the representation $Z$ and Selection Indicator $S$ be conditionally independent, which can not be tested directly because we miss the data for the $S = 0$ part. According to a previous study [15], we can guarantee condition 3 by the following reasonable test, which is shown as follows:

$$E[S/Q(R) \mid Z] = 1 \tag{3}$$

where $Q$ is an arbitrary function, which can be modelled by parametric or non-parametric estimation. In case of $S = 0$, $S/Q(R)$ is constant at 0.

In accordance with Theorem 1, the feasibility of Condition 3 can be evaluated by determining whether there exists a solution $Q \in (0, 1]$ to Eq. (3) for $R$ under the condition $Z$. Satisfying such a solution facilitates the construction of a data distribution in which the shadow variable $Z$ and the collider $S$ adhere to Condition 3.

In practice, we use non-parametric estimation to implement the test in Theorem 1. The *Estimator* module employs an MLP to fit the function $1/Q(R)$, and is equipped with $\mathcal{L}_2$ regularization to mitigate the ill-posed problem of non-parametric estimation. According to the problem formulation in Section III, the observed data are virtually missing for $S = 0$, i.e. $O$ can be taken as an approximate substitution for $S$, which is a typical treatment in previous research on MNAR [34], [36], [43]. If the output of *Estimator* is $\tilde{Q}$, denote $D = \tilde{Q} \times O_u$. Then, based on the BCE loss, Eq. (3) having a solution equivalent to minimising $\mathcal{L}_{Test} = BCE(D, O_u)$.

**Theorem 1:** Suppose $Z \not\perp\!\!\!\perp O \mid X, R, S = 1$, $Z \not\perp\!\!\!\perp R \mid$

**Algorithm 1** Shadow Variable Catcher

**Input:** Exposure $O$, Rating $R$, Covariate $X$
**Output:** The latent representation of shadow variables $Z$

```
Initialize SVC model with parameter η;
while Stop condition is not reached do
    Fetch (O_u,R_u,X_u) from (O, R, X);
    Input O_u and X_u into Encoder to get mean and variance;
    Sample Z through reparameterization tricks;
    Reconstruct Õ_u from Decoder using Z;
    Predict R̃_u from Predictor using Z;
    Get Q̃ via Estimator using R̃_u;
    Minimize L_SVC to optimize η;
end while
Choose the parameter η with the best L_Pred and then generate shadow variables for each user;
return Z
```

$X, O, S = 1$, and the overlap assumption[1] hold. Then $Z \perp\!\!\!\perp S \mid X, O, R$ can be rejected if and only if there exists no solution $Q$ to Eq. (3) that belongs to $(0, 1]$ [15].

The overall structure of the SVC model is shown in the left part of Figure 4. Valid shadow variables (e.g., user interests) can be trained out as close as possible with the above three losses. As all three losses are calculated with BCE loss, the final loss of our SVC is defined as:

$$\mathcal{L}_{SVC} = \mathcal{L}_{Rec} + \mathcal{L}_{Pred} + \mathcal{L}_{Test} + \mathcal{L}_{KL} \tag{4}$$

The constraint losses in $\mathcal{L}_{SVC}$ are inherently collaborative, not conflicting. Therefore, they can be harmoniously co-optimized. Finally, we select the optimal weights that achieve the best performance on $\mathcal{L}_{Pred}$ and generate the latent representation $Z$ of shadow variables for each user. Once we obtain $Z$, we can use it as a shadow variable to remove the selection bias caused by data with MNAR.

### *B. Shadow variable debiased recommendation*

In this section, we introduce our proposed Shadow Debiased Recommendation (SDR). The framework of SDR is depicted in the right part of Figure 4.

The *Odds Ratio* ($OR$) is commonly employed to quantify the influence of a factor on an outcome [44]. Consequently, we utilize $OR$ to encode the disparity between the distributions $(O, X, Z, R, S = 1)$ and $(O, X, Z, R, S = 0)$. Since the shadow variable $Z$ is independent of the selection indicator $S$, the OR of the selection indicator $S$ to $R$ is calculated as follows:

$$\begin{aligned} &OR(R \mid O, X, Z) = OR(R \mid O, X) \\ &= \frac{P(S=0 \mid R, O, X)P(S=1 \mid R=0, O, X)}{P(S=0 \mid R=0, O, X)P(S=1 \mid R, O, X)} \end{aligned} \tag{5}$$

[1]The overlap condition requires that the training and test sets share enough common support areas to ensure that the training set's insights can be generalized to the test set.

where $R = 0$ is a reference value that can be replaced by any value within the range of $R$ values. When $R = 0$ is used as a reference value, it is evident that $OR(R = 0 \mid O, X) = 1$. It must be ensured that $OR(R \mid O, X) > 0$ and $E[OR(R \mid O, X) \mid X, Z, S = 1] < +\infty$. While $OR(R = 0 \mid O, X) = OR(R = 1 \mid O, X) = 1$, it means that there is no disparity between distributions $(O, X, Z, R, S = 1)$ and $(O, X, Z, R, S = 0)$. With the aid of shadow variables, we have the proposition of identification for recovering the missing distribution $P(R = 1 \mid O, X, Z, S = 0)$ according to previous works [18], [19]. We do not present this proposition here due to page limitations.

We are able to recover the missing distributions $P(R = 1 \mid O, X, Z, S = 0)$ by using shadow variables with the following steps. Firstly we calculate $\tilde{OR}$ by:

$$\tilde{OR}(R \mid O, X) = \frac{OR(R \mid O, X)}{E[OR(R \mid O, X) \mid O, X, S = 1]} \tag{6}$$

The distributional disparity $OR$ can be captured by the shadow variables $Z$ as:

$$E[\tilde{OR}(R \mid O, X) \mid O, X, Z, S = 1] = \frac{f(Z \mid O, X, S = 0)}{f(Z \mid O, X, S = 1)} \tag{7}$$

This equation is a Fredholm integral equation of the first kind, with $\tilde{OR}$ to be solved for. Using $OR$, we have the recovery equation for the missing distribution $P(R = 1 \mid O, X, S = 0)$ as follows:

$$\begin{aligned} &P(R = 1 \mid O, X, Z, S = 0) \\ &= \frac{OR(R = 1 \mid O, X)P(R = 1 \mid O, X, Z, S = 1)}{E[OR(R \mid O, X) \mid O, X, Z, S = 1]} \end{aligned} \tag{8}$$

Thus, we have the following theorem for identification of $P(R = 1 \mid O, X, Z, S)$ using shadow variables $Z$.

**Theorem 2 (Identification of $P(R = 1 \mid O, X, Z, S)$ [18]):** Under Assumption 1 and the completeness condition of $P(R = 1 \mid O, X, Z, S = 1)$[2], Eq. (6) has a unique solution. Thus, $OR(R \mid O, X)$ and $P(R = 1 \mid O, X, Z, S)$ can be identified.

Theorem 2 ensures non-parametric identification under MNAR data through shadow variable. The theory involves only the proposed completeness conditions and observational data, and it can be justified without extra model assumptions on the missing data distribution.

**Identification Under Collaborative Filtering.** The purpose of collaborative filtering (CF) is to provide a pre-ranking with a lightweight model, with an emphasis on ranking. Estimating precise $P(R = 1 \mid O, X, Z, S = 1)$ will undermine the performance of the model. To recover the missing distributions under CF, it is necessary to adjust the identification process.

Following the above process, we approximate the probability distribution of $S$ by using the probability distribution of $O$ as a substitute. Based on the approximate substitution $P(S \mid R, O, X) \approx P(O \mid R, X)$, we adjust Eq. (5) as follows:

$$OR(R \mid O, X) = \frac{P(O=0 \mid R, X)P(O=1 \mid R=0, X)}{P(O=0 \mid R=0, X)P(O=1 \mid R, X)} \tag{9}$$

[2]Completeness is widely used in identification problems, requiring the dataset to have sufficient information to train the model. This is linked to both the dataset and the applied model. The public datasets and baseline model used in this paper can be considered to have completeness.

where $P(O \mid R, X)$ can be accessed by either Naive Bayes or Logistic regression [34]. Eq. (9) allows us to estimate the distributional disparity $OR$ from data. Hereafter, based on Eqs. (6), (7), (8) and approximate substitutions, we have:

$$\begin{aligned}
&P(R=1 \mid O, X, Z, S=0) = \\
&\frac{OR(R=1 \mid O, X) P(R=1 \mid O, X, Z, S=1)}{E[OR(R \mid O, X) \mid O, X, Z, S=1]} = \\
&\frac{\tilde{OR}(R=1 \mid O, X) P(R=1 \mid O, X, Z, S=1)}{E[\tilde{OR}(R \mid O, X) \mid O, X, Z, S=1]} = \\
&\frac{f(Z \mid O, X, S=1)}{f(Z \mid O, X, S=0)} \tilde{OR}(R=1 \mid O, X) P(R=1 \mid O, X, Z, S=1) \\
&= \frac{f(Z \mid X, O=1)}{f(Z \mid X, O=0)} \tilde{OR}(R=1 \mid O, X) P(R=1 \mid O, X, Z, S=1)
\end{aligned} \tag{10}$$

Therefore, the last step is the approximate substitution of $O$ for $S$. Eq. (10) is designed to retain as much information as possible, thereby facilitating us to achieve unbiased training of CF model. In our SDR method, we use the learned $Z$ by SVC to train recommendation models without selection bias based on the recovery Eq. (10) and the average causal effect Eq. (1). Distribution $P(R=1 \mid O, X, Z, S=1)$ can be modelled with any collaborative filtering model.

We use Matrix Factorization (MF) as the backbone for our SDR method, modelling based on a simple additive model $f_\varphi(R=1 \mid O, X, Z, S=1) = f_1(u,i) + f_2(z_u, i)$. According to the proposition, for the unbiased model $f_\varphi(R=1 \mid O, X, Z, S)$, we have:

$$\begin{aligned}
&f_\varphi(R=1 \mid O, X, Z, S) \\
&= \sum_{d}^{0,1} f_\varphi(R=1 \mid O, X, Z, S=d) P(O=d \mid X) \\
&= P(O=1 \mid X) f_\varphi(R=1 \mid O, X, Z, S=1) \\
&+ \alpha \cdot P(O=0 \mid X) f_\varphi(R=1 \mid O, X, Z, S=1)
\end{aligned} \tag{11}$$

where $\alpha = \frac{\tilde{OR}(R=1|O,X) f(Z|X,O=1)}{f(Z|X,O=0)}$ and $P(O=1 \mid X)$, $P(O=0 \mid X)$, $\tilde{OR}(R=1 \mid O, X)$ can be estimated from the dataset. The ratio between $f(Z \mid X, O=1)$ and $f(Z \mid X, O=0)$ can be estimated by any available density estimation model. We applied *Real NVP* [45] to estimate the discrepancy between the distributions under $O=1$ and $O=0$. Specifically, *Real NVP* models $h_1(Z)$ and $h_0(Z)$ are respectively trained by optimizing the following negative log-likelihood functions:

$$\mathcal{L}_{Z_1} = \sum_{i:O_i=1} -log(h_1(Z_i)), \quad \mathcal{L}_{Z_0} = \sum_{i:O_i=0} -log(h_0(Z_i)) \tag{12}$$

To ensure the efficiency of the recommender system, we aim to train the unbiased model $f_\varphi(R=1 \mid O, X, Z, S)$ directly to reduce inference time. For this purpose, we adjusted the training loss to be:

$$\mathcal{L}_{SDR} = \sum_{(u,i):O_{u,i}=1} \phi(f_\varphi(R=1 \mid O, X, Z, S=1)/\tilde{\alpha}, r_{u,i}) \tag{13}$$

where $\tilde{\alpha} = P(O=1 \mid X) + \alpha \cdot P(O=0 \mid X)$, and $\phi(\cdot)$ is a *pointwise* loss for recommendation. We use mean squared error (MSE) loss for training our SDR. The process of our proposed SDR is illustrated in the Algorithm 2. The debiasing technique in this work is done at training. Compared to Backbone MF, SDR only requires additional calculation of low-dimensional shadow variables during model inference, which is friendly to real-time recommendations.

**Algorithm 2** Shadow Debiased Recommendation

**Input:** Exposure $O$, Rating $R$, Shadow Variable $Z$
**Output:** Unbiased model $f_\varphi(R=1 \mid O, X, Z, S)$

**Training Phase**
Calculate $\tilde{OR}(R \mid O, X)$ via Eq. (9) and Eq. (6);
Train density estimators $h_1(Z)$, $h_0(Z)$ via Eq. (12);
Initialize a recommender model with parameter $\varphi$;
**while** Stop condition is not reached **do**
  Fetch $(u, i)$ from $O_{u,i}=1$;
  Minimize the loss Eq. (13) to optimize $\varphi$;
**end while**
**return** $f_\varphi(R=1 \mid O, X, Z, S)$ with $\varphi$

**Limitations.** The validity of our SDR rests on the assumption that shadow variables about collision variables can be captured from user interactions $U$ and user features $X$. If certain collision variables do not meet this assumption, the selection bias they introduce may persist, potentially undermining the model's accuracy.

## V. EXPERIMENTS

In this section, we evaluate the effectiveness and robustness of our SDR[3] method from three perspectives.

- **RQ1:** Does our proposed SDR model outperform existing debiased models on real-world datasets?
- **RQ2:** How does the proposed model perform under different strengths of selection bias?
- **RQ3:** What concrete effect does each component contribute?

### A. Experimental Settings

*a) Three Real-world Datasets:* To compare the debiasing ability of the models in recommender systems, we conduct experiments on three real-world datasets: Coat[4], Yahoo!R3[5], and KuaiRand[6]. The details of the datasets are summarised in Table I. Each dataset contains biased training data collected from operating recommender systems and unbiased test data collected from randomized experiments. The unbiased data minimize the introduction of biases such as confounding bias and selection bias, allowing us to test the effectiveness of model debiasing. Coat and Yahoo!R3 ratings range from 1 to 5 stars, while KuaiRand consists of positive or negative samples defined by the signal "IsClick". Based on the design of previous work [31], [32], [36], [46], Coat and Yahoo!R3 regard ratings $\geq 4$ as positive feedback, otherwise negative. We use all biased data as the training set, 30% unbiased data as the validation set, and 70% unbiased data as the test set. We adopt Naive Bayes, drawing 5% of the total unbiased dataset from the validation set for parameter estimation.

[3]https://github.com/KongXinSAMA/ShadowRecommendation
[4]https://www.cs.cornell.edu/~schnabts/mnar/
[5]https://webscope.sandbox.yahoo.com/
[6]https://kuairand.com/

TABLE I
THE DETAILS OF THE DATASETS COAT, YAHOO!R3 AND KUAIRAND.

| Dataset | #User | #Item | #Biased Data | #Unbiased Data |
|---|---|---|---|---|
| Coat | 290 | 300 | 6,960 | 4,640 |
| Yahoo!R3 | 5,400 | 1,000 | 129,179 | 54,000 |
| KuaiRand | 23,533 | 6,712 | 1,413,574 | 954,814 |

*b) Synthetic Data Generation:* To test whether the model captures users' real interests under MNAR data, we generative a synthetic dataset in line with causal Figure 3. The synthetic dataset consists of 1,200 users and 400 items. For each user, there is a one-dimensional explicit feature $X \in \{1, 2, 3, 4, 5\}$. Each value of the feature represents a unique influence. To test the ability of hidden variable learning, we add shadow variables to the synthetic data, which exist as implied confounding variables. Shadow variables are influenced by the explicit feature, and the conditional distribution of $Z$ follows:

$$Z_k \mid X \sim N(\mu_k(X), \sigma_k^2(X)), k \in \{1, 2\} \tag{14}$$

where $Z_k$ represents the k-th dimension of $Z$. For the exposure matrix $O$, its generation follows:

$$\begin{aligned} &O \mid Z, X \sim Bernoulli(c(Z, X)) \\ &c(Z, X) = \beta \cdot sigmoid(ZM_1 e_{zi} + X_e M_2 e_{xi}) \\ &X_e \sim N(\mu_x(X), \sigma_x^2(X)) \end{aligned} \tag{15}$$

where $X_e$ is the effect of $X$, sampled from a Gaussian distribution, and $\beta$ is a hyper-parameter that controls the sparsity of the exposure matrix. $e_{zi}^{2\times1}$ and $e_{xi}^{1\times1}$ are randomly generated item-wise embedding vector. The matrices $M_1^{2\times2}$ and $M_2^{1\times1}$ are samples from a uniform distribution representing association patterns. For each user-item pair, the true rating $r_{ui} = \mathbf{I}(e_u^T e_i + z_u e_{zi} + x_u e_{xi} + \epsilon_{ui})$. The function $\mathbf{I}$ is a normalization function that maps ratings to integer ratings from 1 to 5. $\epsilon_{ui}$ is an *i.i.d.* random noise, making the rating close to the real world scenarios.

Next, we construct the MNAR data caused by the user's selection. We impose a collision variable $S$, which is affected by the covariate $X$ but not by the shadow variable $Z$. After interference by selection bias, the exposure matrix $O_s$ follows:

$$\begin{aligned} &O_s = S \circ O \\ &S \mid X \sim Bernoulli(sigmoid(S_e + X)) \\ &S_e \sim N(\mu_s, \sigma_s^2) \end{aligned} \tag{16}$$

The smaller the collider $S$, the lower the probability of being selected by user, i.e., $O$ has minimal $S = 0$ data. For the true distribution of tests, $P(r_{ui}) = 0.5 + \gamma \cdot (3 - r_{ui})/4$ where $\gamma \in [0, 1)$. $\gamma$ is a hyper-parameter that represents the strength of selection bias. When $\gamma = 0$, the training and test sets have the same distribution. The larger the $\gamma$, the greater the discrepancy between the two distributions.

*c) Evaluation Metrics:* We use two classic metrics in recommendation: NDCG@K and Recall@K. Recall@K measures whether items are correctly included in the Top-K recommended list, while NDCG@K focuses on whether the Top-K list is sorted correctly. In the experiment, the mean and variance with 10 different random seeds at K = 5 are reported. All experiments were conducted in the following hardware environments: 12th Gen Intel(R) Core(TM) i5-12400F, with one GeForce RTX 4060 GPU.

TABLE II
GRID SEARCH RANGE OF HYPER-PARAMETER

| Dataset | Hyper-parameter |
|---|---|
| Coat<br>Yahoo!R3<br>SimData | Learning Rate: {1e-3, 5e-4, 1e-4, 5e-5, 1e-5}<br>Weight Decay: {1e-5, 1e-6}<br>Embedding Size: {256, 512} |
| KuaiRand | Learning Rate: {1e-4, 5e-5, 1e-5, 5e-6, 1e-6}<br>Weight Decay: {1e-5, 1e-6}<br>Embedding Size: {512, 1024} |

*d) Baselines:* We compare our method with existing advanced methods for both de-selection bias and hidden confounding learning, incouding (1) **MF [47].** MF is a classic recommendation model that is widely used in collaborative filtering. We use MF as a backbone to test the improvement of all debiasing methods. (2) **IPS [34].** Inverse Propensity Score (IPS) is a fundamental propensity-based approach to dealing with selection bias. This approach adjusts training loss using a propensity score to alleviate selection bias. (3) **DR [38].** DR is a joint learning method for de-selection bias that skilfully combines propensity-based and imputation-based methods to improve the robustness for addressing selection biases. (4) **RD-IPS [36].** Robust Deconfounder (RD) is a modification of the propensity-based approach. Based on sensitivity analysis, it introduce an uncertainty set to estimate the propensity. We take IPS version to test the effect of the propensity modifications. (5) **InvPref [32].** InvPref applies invariant learning methods [48] to discover the user's invariant preferences from data containing unmeasured confounding biases. We consider this as a method of hidden confounding learning. (6) **DeepDCF-MF [30].** Deep Deconfounder (DeepDCF) is a method of hidden confounding learning. The method learns confounding applying VAE and utilises user features to reduce the variance of the model. We select the MF version to ensure backbone consistency. (7) **IDCF [31].** IDCF is an improvement of hidden confounding learning. This approach learned hidden confounding with IVAE, which theoretically guarantees the identifiability of the hidden confounding, enhanced ability to learn unmeasured confounding.

*e) Hyper-parameter Search:* We use grid search to select hyper-parameters according to their performance on the validation set. For the larger dataset KuaiRand, we use a smaller learning rate and a larger embedding space. The range of the grid search is given in Table II. For Hyper-parameters that are unique to the model which are available in the code.

### *B. Performance Comparison (RQ1)*

The experimental results on the three real-world datasets are reported in Table III and we have the following consequences:

TABLE III
RECOMMENDATION PERFORMANCES ON COAT, YAHOO!R3, AND KUAIRAND. THE P-VALUE IS CALCULATED UNDER T-TEST BETWEEN SDR AND THE BEST BASELINE.

| Datasets | Coat | | Yahoo!R3 | | KuaiRand | |
|---|---|---|---|---|---|---|
| | NDCG@5 | RECALL@5 | NDCG@5 | RECALL@5 | NDCG@5 | RECALL@5 |
| MF | 0.5616 ± 0.0137 | 0.5532 ± 0.0180 | 0.5809 ± 0.0067 | 0.7228 ± 0.0090 | 0.3788 ± 0.0006 | 0.3291 ± 0.0006 |
| IPS | 0.5738 ± 0.0092 | 0.5557 ± 0.0098 | 0.5669 ± 0.0032 | 0.7096 ± 0.0043 | 0.3736 ± 0.0008 | 0.3262 ± 0.0009 |
| DB | 0.5715 ± 0.0175 | 0.5520 ± 0.0206 | 0.5660 ± 0.0045 | 0.7100 ± 0.0065 | 0.3727 ± 0.0015 | 0.3252 ± 0.0013 |
| RD-IPS | 0.5777 ± 0.0105 | 0.5598 ± 0.0127 | 0.5683 ± 0.0041 | 0.7115 ± 0.0048 | 0.3755 ± 0.0034 | 0.3256 ± 0.0022 |
| InvPref | 0.5719 ± 0.0128 | 0.5502 ± 0.0173 | 0.6096 ± 0.0034 | 0.7519 ± 0.0044 | 0.3828 ± 0.0009 | 0.3306 ± 0.0006 |
| DeepDCF-MF | 0.5804 ± 0.0116 | 0.5683 ± 0.0096 | 0.6315 ± 0.0039 | 0.7680 ± 0.0046 | 0.4059 ± 0.0017 | 0.3465 ± 0.0019 |
| IDCF | 0.5825 ± 0.0057 | 0.5704 ± 0.0090 | 0.6523 ± 0.0016 | 0.7822 ± 0.0033 | 0.4109 ± 0.0010 | 0.3512 ± 0.0011 |
| SDR(ours) | **0.5964** ± 0.0090 | **0.5776** ± 0.0095 | **0.6702** ± 0.0014 | **0.7952** ± 0.0017 | **0.4254** ± 0.0004 | **0.3610** ± 0.0004 |
| p-value | $1e^{-3}$ | $1e^{-1}$ | $1e^{-14}$ | $5e^{-9}$ | $1e^{-19}$ | $4e^{-15}$ |

- Our proposed SDR model achieves the best prediction accuracy on all datasets with excellent mean and variance. All reported p-values indicate statistical significance, except for Coat's Recall@5 metrics, as it is a small dataset and cannot delicately test the model. This result demonstrates the effectiveness of learning shadow variables and unbiased training for debiasing, which we will further validate in subsequent experiments.
- IPS, DB, and RD-IPS were designed to address selection bias but did not consider the impact of confounding bias, resulting in suboptimal performance. In contrast, SDR captured shadow variables are used as proxies for hidden confounding variables to mitigate confounding bias, in addition to addressing selection bias.
- In the smaller dataset, Coat, SDR performs similarly to IDCF and DCF, but as the size of the dataset increases, our model shows superior performance, indicating that SDR fully exploits the information in the dataset to capture the real interests of users.

### C. Experiments on Synthetic Data (RQ2)

We verify whether the proposed method captures the real interests of users under MNAR data. We generate a synthetic dataset which is adjusted by three main hyper-parameters. The parameter $\beta \in (0, 1]$ controls the sparsity of the exposure matrix; a smaller value of $\beta$ results in a sparser matrix. $\gamma \in [0, 1)$ represents the strength of the selection bias. When $\gamma = 0$, the data has no selection bias. As $\gamma$ increases, the strength of the selection bias increases. $\epsilon$ is the random noise in the exposure matrix used to simulate error exposure.

*a) De-selection Bias Experiments:* Experiments were conducted with varying strengths of selection bias on $\beta = 0.1$ and $\epsilon = 10$. Experimental results are reported in IV, and we have the following observations:

- At $\gamma = 0$, the dataset exhibits no discernible selection bias, with only a minimal degree of confounding bias. IDCF achieves the best baseline due to its effective handling of confounding bias. SDR achieves similar results to IDCF as it is also capable of addressing confounding bias.
- At $\gamma = 0.8$, selection bias dominates the dataset. In this case, methods for dealing with selection bias such as IPS-RD, RD, and IPS achieve results similar to the advanced IDCF. Our SDR performs optimally because it mitigates both selection bias and confounding bias.
- DeepDCF performs poorly, which is attributed to the direct introduction of features within the context of complex feature patterns. SDR and IDCF utilize explicit features only indirectly and are not affected by complex features.

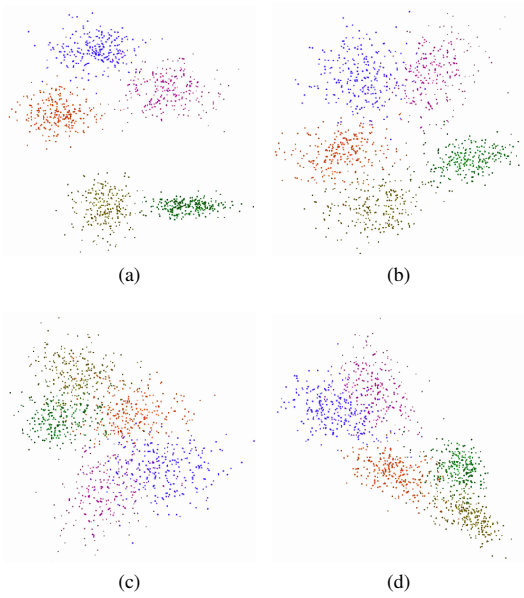

Fig. 5. Visualisation of implicit variable learning. (a) The true distribution. The distribution estimated by (b) SVC, (c) IDCF, (d) DeepDCF.

*b) Implicit Variable Learning:* Furthermore, we test the model's ability to learn implicit variables on synthetic data. To show clear results, we conduct experiments under $\beta = 0.8$ and $\epsilon = 0$. As shown in Figure 5: (a) shows the true distribution of shadow variables, (b) presents our SVC, and (c) and (d) indicates the two models IDCF and DeepDCF.

In this experiment, we observe that: (1) The SVC, based on elaborate constraints, achieves the clearest distinction. (2) IDCF utilizes explicit features, thus achieving sub-optimal results with insufficient distinction between distributions. (3) DeepDCF is superior to IDCF in terms of distinction but overly clustered in distribution. This justifies the ability of

TABLE IV
EXPERIMENTAL RESULTS UNDER DIFFERENT STRENGTHS OF SELECTION BIAS. THE BEST AND ROUND-UP RESULTS ARE HIGHLIGHTED AND UNDERLINE, RESPECTIVELY.

| Datasets | $\gamma$=0 | | $\gamma$=0.8 | |
|---|---|---|---|---|
| | NDCG@5 | RECALL@5 | NDCG@5 | RECALL@5 |
| MF | 0.8793 ± 0.0025 | 0.8683 ± 0.0021 | 0.8359 ± 0.0035 | 0.9457 ± 0.0024 |
| IPS | 0.8791 ± 0.0059 | 0.8696 ± 0.0038 | 0.8384 ± 0.0078 | 0.9471 ± 0.0033 |
| DB | 0.8776 ± 0.0043 | 0.8667 ± 0.0029 | 0.8421 ± 0.0010 | 0.9444 ± 0.0009 |
| IPS-RD | 0.8766 ± 0.0064 | 0.8684 ± 0.0038 | 0.8436 ± 0.0033 | 0.9461 ± 0.0032 |
| InvPref | 0.8756 ± 0.0044 | 0.8678 ± 0.0032 | 0.8366 ± 0.0041 | 0.9456 ± 0.0041 |
| DeepDCF-MF | 0.8447 ± 0.0031 | 0.8394 ± 0.0029 | 0.7967 ± 0.0057 | 0.9070 ± 0.0051 |
| IDCF | 0.8820 ± 0.0024 | 0.8709 ± 0.0027 | 0.8444 ± 0.0028 | 0.9482 ± 0.0023 |
| SDR(ours) | **0.8837** ± 0.0019 | **0.8731** ± 0.0018 | **0.8546** ± 0.0021 | **0.9527** ± 0.0027 |
| p-value | $1e^{-1}$ | $5e^{-2}$ | $1e^{-7}$ | $1e^{-3}$ |

SVC to mine implicit shadow variables from observational recommendation data.

### D. In-depth Analysis (RQ3)

Next we will analyse precisely the contribution of each component in our SDR.

*a) SDR Ablation Study:* In SDR, a sequential process involves learning shadow variables and then proceeding to unbiased training. Table V presents the results of this ablation experiment on the Yahoo! R3 dataset. We report the outcomes on both the validation and test sets, along with the relative improvement (RI) of the model over the baseline Matrix Factorization (MF). The Matrix Factorization With Variables (MF-WV) model incorporates only the learned shadow variables $Z$ and does not include the unbiased training module.

TABLE V
SDR ABLATION EXPERIMENTS ON THE YAHOO!R3 DATASET. RI REFERS TO THE RELATIVE IMPROVEMENT OF MF-WV AND SDR OVER THE BASELINE MF.

| Method | NDCG@5 | | RI | RECALL@5 | | RI |
|---|---|---|---|---|---|---|
| | Val | Test | | Val | Test | |
| MF | 0.5774 | 0.5809 | - | 0.7259 | 0.7228 | - |
| MF-WV | 0.6489 | 0.6615 | 13.87% | 0.7844 | 0.7894 | 9.21% |
| SDR | 0.6563 | 0.6702 | 15.37% | 0.7871 | 0.7952 | 10.01% |

It can be observed that the addition of the shadow variables is the most significant improvement to the model. This suggests that learned shadow variables are of enormous value in debasing. And with the addition of unbiased training which does not incur the cost of model inference, the model is clearly enhanced in both validation and testing.

*b) SVC Ablation Study:* We conducted an in-depth study of the impact of the loss components of SVC. Table VI presents the test set performance of different loss-learned variables on model MF-WV in the Yahoo!R3 dataset. Upon analysis of the experiments, the following findings emerged:

- SVC learned variables perform worst when w/o $\mathcal{L}_{Rec}$. Since under the experimental setup (Section V-A), the rating matrix contained less information, it is not possible to guarantee the source of the latent variables.
- The loss $\mathcal{L}_{Rec}$ ensures that sufficient information is available to generate the shadow variables. While $\mathcal{L}_{Pred}$ and $\mathcal{L}_{Test}$ are two auxiliary losses that constrain the generated variables into required shadow variables for recommendation debiasing.
- The loss $\mathcal{L}_{Test}$ has the least contribution to the SVC. Despite not enhancing the model accuracy markedly, $\mathcal{L}_{Test}$ empirically adjusts the distribution of $Z$ to align with the Condition 3 for shadow variables.

TABLE VI
TEST RESULTS OF DIFFERENT LOSS-LEARNED VARIABLES ON MF-WV IN THE YAHOO!R3 DATASET. FROM TOP TO BOTTOM REPRESENTS THE IMPORTANCE OF W/O LOSSES.

| Loss | NDCG@5 | RECALL@5 |
|---|---|---|
| w/o $\mathcal{L}_{Rec}$ | 0.5840 | 0.7299 |
| w/o $\mathcal{L}_{Pred}$&$\mathcal{L}_{Test}$ | 0.6523 | 0.7822 |
| w/o $\mathcal{L}_{Pred}$ | 0.6548 | 0.7820 |
| w/o $\mathcal{L}_{Test}$ | 0.6596 | 0.7867 |
| SVC | 0.6615 | 0.7894 |

## VI. CONCLUSION

In this work, we have developed a method to combat selection bias in collaborative filtering. The proposed SVC model effectively learns shadow variables (e.g., user interests) from the data. The shadow variable, as a special covariate, opens up a new causal path between exposure and rating. Based on causal inference techniques, we have also proposed a SDR method for addressing selection bias. The SDR framework recovers the missing distribution due to selection bias by utilizing shadow variables, thereby facilitating the training of unbiased recommendation models with average effects. Extensive experiments have validated the effectiveness and robustness of our proposed SDR method in mitigating selection bias in recommender systems. In our future work, we will incorporate real-time user feedback and preferences to dynamically adjust shadow variables and improve recommendation accuracy.

## VII. ACKNOLEDGEMENT

This work was partially supported by the Specific Research Project of Guangxi for Research Bases and Talents (GuiKe AD24010011) and the Australian Research Council (grant number: DP230101122).

## REFERENCES


[1] S. Zhang, X. Li *et al.*, "Learning k for knn classification," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 8, no. 3, pp. 1–19, 2017.

[2] X. Zhu, S. Zhang *et al.*, "Low-rank sparse subspace for spectral clustering," *IEEE Transactions on knowledge and data engineering*, vol. 31, no. 8, pp. 1532–1543, 2018.

[3] W. Fan, Y. Ma *et al.*, "Graph neural networks for social recommendation," in *The world wide web conference*, 2019, pp. 417–426.

[4] W. Lan, T. Yang *et al.*, "Multiview subspace clustering via low-rank symmetric affinity graph," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[5] J. Chen, H. Dong *et al.*, "Bias and debias in recommender system: A survey and future directions," *ACM Transactions on Information Systems*, vol. 41, no. 3, pp. 1–39, 2023.

[6] S. Li, Q. Chen, Z. Liu, S. Pan, and S. Zhang, "Bi-sgtar: A simple yet efficient model for circrna-disease association prediction based on known association pair only," *Knowledge-Based Systems*, vol. 291, p. 111622, 2024.

[7] T. Wei, F. Feng *et al.*, "Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 1791–1800.

[8] C. Hansen, C. Hansen *et al.*, "Contextual and sequential user embeddings for large-scale music recommendation," in *Proceedings of the 14th ACM Conference on Recommender Systems*, 2020, pp. 53–62.

[9] C. Gao, S. Li *et al.*, "Kuairand: an unbiased sequential recommendation dataset with randomly exposed videos," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 3953–3957.

[10] C. Gao, Y. Zheng *et al.*, "Causal inference in recommender systems: A survey and future directions," *ACM Transactions on Information Systems*, vol. 42, no. 4, pp. 1–32, 2024.

[11] J. Pearl, *Causality*. Cambridge university press, 2009.

[12] J. M. Hernández-Lobato *et al.*, "Probabilistic matrix factorization with non-random missing data," in *International conference on machine learning*. PMLR, 2014, pp. 1512–1520.

[13] H. Steck, "Evaluation of recommendations: rating-prediction and ranking," in *Proceedings of the 7th ACM conference on Recommender systems*, 2013, pp. 213–220.

[14] M. Wang, M. Gong *et al.*, "Modeling dynamic missingness of implicit feedback for recommendation," *Advances in neural information processing systems*, vol. 31, 2018.

[15] X. d'Haultfoeuille, "A new instrumental method for dealing with endogenous selection," *Journal of Econometrics*, vol. 154, no. 1, pp. 1–15, 2010.

[16] F. Elwert and C. Winship, "Endogenous selection bias: The problem of conditioning on a collider variable," *Annual review of sociology*, vol. 40, pp. 31–53, 2014.

[17] J. Zhao and J. Shao, "Semiparametric pseudo-likelihoods in generalized linear models with nonignorable missing data," *Journal of the American Statistical Association*, vol. 110, no. 512, pp. 1577–1590, 2015.

[18] W. Miao, L. Liu *et al.*, "Identification, doubly robust estimation, and semiparametric efficiency theory of nonignorable missing data with a shadow variable," *arXiv preprint arXiv:1509.02556*, 2015.

[19] W. Miao and E. J. Tchetgen Tchetgen, "On varieties of doubly robust estimators under missingness not at random with a shadow variable," *Biometrika*, vol. 103, no. 2, pp. 475–482, 2016.

[20] W. Li, W. Miao, and E. Tchetgen Tchetgen, "Non-parametric inference about mean functionals of non-ignorable non-response data without identifying the joint distribution," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 85, no. 3, pp. 913–935, 2023.

[21] B. Marlin, R. S. Zemel *et al.*, "Collaborative filtering and the missing at random assumption," *arXiv preprint arXiv:1206.5267*, 2012.

[22] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[23] Z. Liu, Q. Chen, W. Lan, H. Lu, and S. Zhang, "Ssldti: A novel method for drug-target interaction prediction based on self-supervised learning," *Artificial Intelligence in Medicine*, vol. 149, p. 102778, 2024.

[24] D. Cheng, J. Li *et al.*, "Data-driven causal effect estimation based on graphical causal modelling: A survey," *ACM Computing Surveys*, vol. 56, no. 5, pp. 1–37, 2024.

[25] W. Wang, F. Feng *et al.*, "Deconfounded recommendation for alleviating bias amplification," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 1717–1725.

[26] Y. Zhang, F. Feng *et al.*, "Causal intervention for leveraging popularity bias in recommendation," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 11–20.

[27] X. Zhu, Y. Zhang *et al.*, "Mitigating hidden confounding effects for causal recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[28] S. Xu, J. Tan *et al.*, "Deconfounded causal collaborative filtering," *ACM Transactions on Recommender Systems*, vol. 1, no. 4, pp. 1–25, 2023.

[29] H. Liu, D. Tang *et al.*, "Rating distribution calibration for selection bias mitigation in recommendations," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 2048–2057.

[30] Y. Zhu, J. Yi *et al.*, "Deep causal reasoning for recommendations," *ACM Transactions on Intelligent Systems and Technology*, 2022.

[31] Q. Zhang, X. Zhang *et al.*, "Debiasing recommendation by learning identifiable latent confounders," in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 3353–3363.

[32] Z. Wang, Y. He *et al.*, "Invariant preference learning for general debiasing in recommendation," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 1969–1978.

[33] H. Li, K. Wu *et al.*, "Removing hidden confounding in recommendation: a unified multi-task learning approach," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23. Red Hook, NY, USA: Curran Associates Inc., 2024.

[34] T. Schnabel, A. Swaminathan *et al.*, "Recommendations as treatments: Debiasing learning and evaluation," in *international conference on machine learning*. PMLR, 2016, pp. 1670–1679.

[35] Y. Saito, "Asymmetric tri-training for debiasing missing-not-at-random explicit feedback," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 309–318.

[36] S. Ding, P. Wu *et al.*, "Addressing unmeasured confounder for recommendation with sensitivity analysis," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 305–315.

[37] Y. Liu, J.-N. Yen *et al.*, "Practical counterfactual policy learning for top-k recommendations," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 1141–1151.

[38] X. Wang, R. Zhang *et al.*, "Doubly robust joint learning for recommendation on data missing not at random," in *International Conference on Machine Learning*. PMLR, 2019, pp. 6638–6647.

[39] H. Li, C. Zheng, and P. Wu, "Stabledr: Stabilized doubly robust learning for recommendation on data missing not at random," 2023. [Online]. Available: https://arxiv.org/abs/2205.04701

[40] M. B. Mathur and I. Shpitser, "Simple graphical rules to assess selection bias in general-population and selected-sample treatment effects."

[41] D. Cheng, Z. Xu *et al.*, "Causal inference with conditional instruments using deep generative models," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 6, 2023, pp. 7122–7130.

[42] I. Khemakhem, D. Kingma *et al.*, "Variational autoencoders and nonlinear ica: A unifying framework," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2207–2217.

[43] X. Wang, R. Zhang *et al.*, "Combating selection biases in recommender systems with a few unbiased ratings," in *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 2021, pp. 427–435.

[44] M. Szumilas, "Explaining odds ratios," *Journal of the Canadian academy of child and adolescent psychiatry*, vol. 19, no. 3, p. 227, 2010.

[45] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[46] Y. Wang, D. Liang *et al.*, "Causal inference for recommender systems," in *Proceedings of the 14th ACM Conference on Recommender Systems*, 2020, pp. 426–431.

[47] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[48] M. Arjovsky, L. Bottou *et al.*, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.