Causal Inference in Recommender Systems: A Survey and Future Directions

CHEN GAO, Beijing National Research Center for Information Science and Technology, Tsinghua University, China
YU ZHENG, Department of Electronic Engineering, Tsinghua University, China
WENJIE WANG, School of Computing, National University of Singapore, Singapore
FULI FENG, School of Information Science and Technology, University of Science and Technology of China, China
XIANGNAN HE and YONG LI, Department of Electronic Engineering, Tsinghua University, China

Recommender systems have become crucial in information filtering nowadays. Existing recommender systems extract user preferences based on the correlation in data, such as behavioral correlation in collaborative filtering, feature-feature, or feature-behavior correlation in click-through rate prediction. However, unfortunately, the real world is driven by *causality*, not just correlation, and correlation does not imply causation. For instance, recommender systems might recommend a battery charger to a user after buying a phone, where the latter can serve as the cause of the former; such a causal relation cannot be reversed. Recently, to address this, researchers in recommender systems have begun utilizing causal inference to extract causality, thereby enhancing the recommender system. In this survey, we offer a comprehensive review of the literature on causal inference-based recommendation. Initially, we introduce the fundamental concepts of both recommender system and causal inference as the foundation for subsequent content. We then highlight the typical issues faced by non-causality recommender system. Following that, we thoroughly review the existing work on causal inference-based recommender systems, based on a taxonomy of three-aspect challenges that causal inference can address. Finally, we discuss the open problems in this critical research area and suggest important potential future works.

CCS Concepts: • **Information systems** → **Information retrieval**; **Recommender systems;**

Additional Key Words and Phrases: Recommender systems; causal inference; information retrieval

ACM Reference Format:
Chen Gao, Yu Zheng, Wenjie Wang, Fuli Feng, Xiangnan He, and Yong Li. 2024. Causal Inference in Recommender Systems: A Survey and Future Directions. *ACM Trans. Inf. Syst.* 42, 4, Article 88 (February 2024), 32 pages. https://doi.org/10.1145/3639048

This work is supported in part by National Key Research and Development Program of China under 2020AAA0106000, and by National Natural Science Foundation of China under 62272262 and U23B2030. This work is also supported by grant from the Guoqiang Institute, Tsinghua University.
Authors' addresses: C. Gao, Beijing National Research Center for Information Science and Technology, Tsinghua University, China; e-mail: chgao96@gmail.com; Y. Zheng and Y. Li, Department of Electronic Engineering, Tsinghua University, China; e-mails: y-zheng19@mails.tsinghua.edu.cn, liyong07@tsinghua.edu.cn; W. Wang, School of Computing, National University of Singapore, Singapore; e-mail: wenjiewang96@gmail.com; F. Feng and X. He, School of Information Science and Technology, University of Science and Technology of China, China; e-mails: {fulifeng93, xiangnanhe}@gmail.com.

ACM 1046-8188/2024/02-ART88
https://doi.org/10.1145/3639048

1 INTRODUCTION

In the era of information overload, **recommender systems** (**RecSys**) have emerged as the fundamental service for facilitating users' information access. From the early shallow models [47, 75] to recent advances of deep learning-based ones [15, 31] and the most recent graph neural network-based models [29, 131], the techniques and models of recommender systems are developing rapidly. In general, recommender systems aim to learn user preferences by fitting historical behaviors, along with collected user profiles, item attributes, or other contextual information. Here, the interaction is mainly induced by the previous recommender system and is largely affected by the recommendation policy. Then, recommender systems filter from the item-candidate pools and select items that match users' personalized preferences and demands. Once deployed, the system collects new interactions to update the model, where the whole framework thus forms a feedback loop.

Generally, recommender systems can be divided into two categories: **collaborative filtering** (**CF**) and content-based recommendation (*a.k.a.*, **click-through rate** (**CTR**) prediction, shortened as CTR prediction). Collaborative filtering focuses on users' historical behaviors, such as clicking, purchasing, and so on. The basic assumption of collaborative filtering is that users with similar historical behaviors tend to have similar future behaviors. For example, the most representative **matrix factorization model** (**MF**) uses vectors to represent users and items, and then it uses the inner product to calculate the relevance scores between users and items. To improve the model capacity, recent work [15, 31] takes advantage of deep neural networks for matching users with items, such as neural collaborative filtering [31], which leverages multi-layer perceptrons to replace the inner product in the MF model. Furthermore, a broad view of collaborative filtering models the relevance with consideration of additional information, such as the timestamp of each behavior in sequential recommendation [12, 132], user social network in social recommendation [17, 114], and multi-type behaviors in multi-behavior recommendation [21, 117], and so on. CTR prediction focuses on leveraging the rich attributes and features of users, items, or context to enhance recommendation. The mainstream CTR prediction task aims to learn high-order features with the proper feature-interaction module, such as the linear inner product in **Factorization Machine** (**FM**), multi-layer perceptrons in DeepFM [24], attention networks in AFM [119], stacked self-attention layers in AutoInt [91], and so on.

The basis of today's recommender systems is to model the *correlation*, such as behavioral correlation in collaborative filtering, feature-feature, or feature-behavior correlation in click-through rate prediction. However, the real world is driven by *causality* rather than correlation, while correlation does not imply causation. Two kinds of causality widely exist in recommender systems, user-aspect, and interaction-aspect. The user-aspect causality refers to the users' decision process being driven by causality. For example, a user may buy a battery charger after buying a phone, in which the latter can serve as the cause of the former, and such a causal relation cannot be reversed. The interaction-aspect causality refers to that the recommendation strategy largely affects users' interactions with the system. For example, the unobserved user-item interaction does not mean that the user does not like the item, which may only be caused by non-exposure.

Formally speaking, causality can be defined as *cause* and *effect* in which the cause is partly responsible for the effect [128]. Causal inference is defined as the process of determining and further leveraging the causal relation based on experimental data or observational data [128]. Two popular and widely-used causal-inference frameworks are the potential outcome framework (Rubin Causal Model) [76], and the **structural causal model** (**SCM**) [69, 71]. Rubin's Framework aims to calculate the effect of certain treatments. The structural causal model establishes a causal graph and corresponding structural equations, comprising a set of variables and structural equations that depict the causal relationships between these variables.

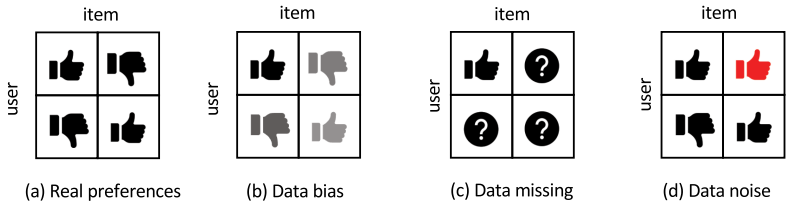

Fig. 1. A simple comparison among three kinds of data issues: data bias, data missing, and data noise, taking collaborative filtering as an example. In general, data bias refers to the biased data collection (e.g., in conformity bias, user behavior does not full reflect preferences as it may be due to conformity); data missing refers to the unobserved preferences (labeled with question marks); data noise refers to incorrect data (marked with red color). As a simple illustration, this figure does not cover other recommendation tasks.

Since following a correlation-driven paradigm, existing recommender systems still suffer from critical bottlenecks. Specifically, three main challenges limit the effectiveness of the current paradigm, for which causal inference can serve as a promising solution, as follows:

— **The issues of data bias**. The collected data, such as the most important user-item interaction data, is observational (not experimental), resulting in biases including conformity bias, popularity bias, and so on. [55] As for the non-causality recommender systems, not only the desired user preferences but also the data bias are learned by the model, which leads to inferior recommendation performance.

— **The issues of data missing or even data noise.** The collected data in recommender systems is limited by the collection procedure, which makes there is missing or noisy data. For example, despite the large-scale item pool, the users only interact with a tiny fraction of items, which means plenty of unobserved user-item feedback cannot be collected. Moreover, sometimes the observed implicit feedback is even noisy, not reflecting the actual satisfaction of users, such as those click behaviors that end with negative reviews on E-Commerce websites or some behaviors by mistake.

— **The beyond-accuracy objectives are hard to achieve.** Besides accuracy, recommender systems should also consider other objectives, such as fairness, explainability, transparency, and so on. Improving these beyond-accuracy objectives may hurt the recommendation accuracy, resulting in a dilemma. For example, a model that considers the *multiple driven causes under user behavior*, based on assigning each cause with disentangled and interpretable embedding, can well provide both accurate and explainable recommendation. Another important objective is diversity but a high-diversity item recommendation list may not be able to well fit user interest. Here causal inference can help capture *why users consume specific category* of items, achieving both high accuracy and diversity.

Recent research on recommender systems tackles these challenges with carefully-designed causality-driven methods. Over the last two years, there has been a surge of relevant articles, and there is a very high probability that causal inference will become predominant in the field of recommender systems. In this survey article, we systematically review these pioneering research efforts, especially focusing on how they address the critical shortcomings with causal inference.

First, recommendation methods incorporating causality can construct a causal graph. Within this framework, bias is typically viewed as a confounder, which can then be addressed using causal-inference techniques. Second, regarding the issue of missing data, causality-enhanced models can assist in constructing a counterfactual world. Thus, the missing data can be inferred through counterfactual reasoning. Third, causal inference naturally facilitates the development of interpretable and controllable models. As a result, the explainability of both the model itself and

the recommendation outcomes can be enhanced. Moreover, other objectives, such as diversity and fairness, can also be realized since the model becomes more controllable. Specifically, the current works of causal inference in recommendation can be categorized as follows:

— **Data debiasing with causal inference.** For issues like popularity bias or exposure bias, the bias (arising from popularity-aware or exposure strategy-aware data collection) can often be seen as a form of confounder. Some existing work addresses this through backdoor adjustment. Conformity bias, on the other hand, can be conceptualized as a collider effect.
— **Data augmentation and data denoising with causal inference.** The dual challenge of data missing encompasses both limited user-data collection and the recommendation model's causal effect on the system. The extreme form of the first challenge can even lead to data noise. For the first challenge, counterfactual reasoning can be employed to generate the uncollected data as augmentation, thus addressing the data-missing problem. For the latter, causal models like IPW can be utilized to estimate the causal impact of recommendation models.
— **Achieving explainability, diversity, and fairness via interpretable and controllable recommendation models using causal inference.** Models crafted in alignment with the causal graph are intrinsically controllable. Some notable techniques in this regard encompass causal discovery and disentangled representations. Leveraging the interpretable model, high diversity can be realized by manipulating the model to sidestep the tradeoff, and fair recommendations can be secured by steering the model to ensure fairness across specific user demographics.

It is worth mentioning that although there are surveys on either recommender systems [25, 113, 134] or causal inference [26, 63, 63, 129], there is no existing survey fully discussing this new and important area of causality-driven recommender systems. Note that there is a very short article (8 pages) [116] trying to survey existing work of causal-inspired recommendation methods, but it only discusses a few of representative articles due to its page limit. These surveys on recommender systems mainly introduce and discuss the basic concepts and various advances of recommender systems, with only a few discussions on causality-based recommendation. On the other hand, surveys of causal inference primarily introduce and discuss the basic concepts and fundamental methods of causal inference, lacking sufficient discussions on applications.

There is a survey [10] about bias and debias in recommender system and we would discuss its relations with our survey as follows. First, the survey [10] concentrates on the bias issue in recommendations and describes how various works address these issues. Among these, causal inference-based methods represent just one segment, with numerous other methods available for tackling bias. Similarly, our survey underscores that while using causal inference to address data bias is a significant component, it is merely a portion of our broader theme: causal inference for recommender systems. Hence, even though some overlap exists between the two surveys, it is small due to the distinct focal topics. Second, when considering the shared part, the two surveys adopt different manners to discuss existing works. Our survey places greater emphasis on the causal inference technique itself, its ties to conventional causal inference methods, and its relevance to other challenges, such as data missing and data noise. In contrast, the bias survey [10] delves deeper into the intricacies of biases (types, origins, etc.) and elaborates on how causal inference-based methods differentiate themselves from other kinds of methods.

We summarize the contribution of this survey as follows:

— To the best of our knowledge, we take the pioneering step to give a systematic survey of this new yet promising area. We categorize the existing work by answering the fundamental

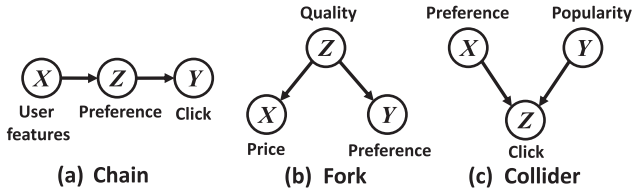

Fig. 2. Illustration of three typical DAGs.

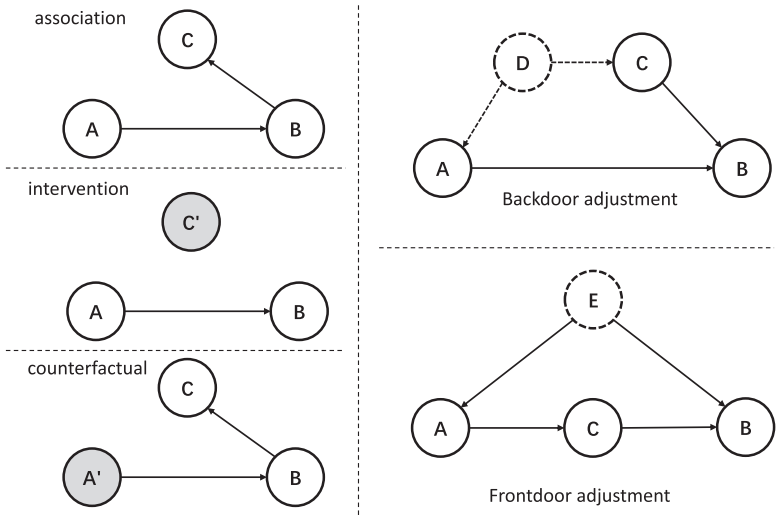

Fig. 3. Important concepts of causal inference.

question of *why the causal inference is needed and how causal inference enhances recommendation.*

— We first provide the necessary knowledge of recommender systems and causal inference. Subsequently, we introduce and explain the existing work of causal inference for recommendation, from the early attempts to the recently-published articles until 2023.

— We discuss important yet unresolved problems in this research area and propose promising directions, which we believe will be the mainstream research direction of the next few years.

2 BACKGROUND

As a survey of the interdisciplinary area of causal inference and recommender systems, we first introduce the background knowledge and fundamental concepts of these two topics.

2.1 Causal Inference

We introduce the fundamental concepts of causal inference to facilitate the readers' understanding. This involves two representative causal frameworks: SCMs proposed by Pearl et al. [71] and the potential outcome framework developed by Rubin et al. [76]. Considering the topic of this survey, we will elaborate on the core concepts using examples from recommender systems for clearer understanding. The basic concepts are shown in Figure 2 and Figure 3, which we will explain in detail in the following sections.

2.1.1 Structural Causal Models. Generally, SCMs abstract the causal relationships between variables into causal graphs, build structural functions, and then conduct causal inference to estimate the effects of interactions or counterfactuals [71].

Causal Models. Causal models involve two essential concepts: causal graphs and structural functions. Specifically, a causal graph describes the causal relationships via a **Directed Acyclic Graph** (**DAG**), in which the nodes denote variables and the edges indicate causal relationships. According to a causal graph, structural functions are used to model the relationships. For each variable, one structural function calculates its value based on its parent nodes.

Three Typical DAGs. As shown in Figure 2, there are three classic structures in causal graphs: *chain*, *fork*, and *collider*, for each of which we give an example of recommender systems. In the chain structure, X affects Y via the mediator Z. For example, in Figure 2(a), the user features affect the user preferences, and the user preferences affect the users' click behavior. Besides, in the fork structure, Z is a confounder, affecting both X and Y. For example, as shown in Figure 2(b), an item's quality can affect both its price and users' preferences toward it. In such a fork structure, Z is defined as *confounder variable*. Roughly ignoring confounder Z leads to **spurious** correlation between X and Y. That is, products with higher prices may have larger sales on an e-commerce platform, which does not mean users prefer to spend much money. In Figure 2(c), differently, Z represents a collider, which is affected by X and Z. For example, the users' click behavior is affected by user preference and item popularity. Conditioning on Y will lead to **correct** correlation between X and Z. That is, users' behaviors on two items with the same popularity level are only affected by their preferences.

Intervention. Given the causal graph, a basic concept of intervention can be formally defined. Specifically, the intervention on a variable X is formulated with do-calculus, $do(X = x)$ [71], which blocks the effect of X's parents and set the value of X as x. For example, $do(X = x)$ in Figure 2(b) will rule out the path $Z \rightarrow X$ and force X to be x [72]. That is, in our above-mentioned example, we set the item prices to a specific value.

Counterfactual. Another important concept is the counterfactual, which contrasts with the factual. It is used to address scenarios where the treatment variable's value settings do not occur in the real world. In other words, counterfactual inference estimates what the outcome would have been if the treatment variable had taken on a different value compared to its observed value in the real world [71]. For example, a bankrupted seller might wonder about potential sales in a counterfactual world where he/she had purchased advertisement services, setting the treatment variable $T_{\textbf{if\_ads}} = 1$.

2.1.2 Potential Outcome Framework. The potential outcome framework [76] is another widely-used causal inference framework besides the structural causal model [71]. It estimates the causal effect of a treatment variable on an outcome variable without the need for a causal graph.

Potential Outcome [76]. Given the treatment variable T and the outcome variable Y, the potential outcome Y_t^i denotes the value of Y under the treatment $T = t$ for individual i. In the factual world, we can only observe the potential outcome of Y under one treatment for each individual.

Treatment Effect [76]. Given binary treatments $T = 0$ or 1, the **Individual Treatment Effect** (**ITE**) for an individual i is defined as $Y_1^i - Y_0^i$. However, ITE is impossible to calculate since we can only observe one potential outcome. Hence, ITE is extended to **Average Treatment Effect** (**ATE**) over a population. For a population $i = \{1, 2, \ldots, N\}$, ATE is calculated by $\mathbb{E}_i \left[Y_1^i - Y_0^i\right] = \frac{1}{N} \sum_{i=1}^{N} \left(Y_1^i - Y_0^i\right)$.

Discussions about these two frameworks. We briefly summarize the similarities and differences between the two frameworks. As stated by Pearl [70], the two frameworks are logically equivalent. The theorem and assumptions in one framework can be equivalently translated into

the language of the other framework. However, the key difference is that the potential outcome framework neither considers the causal graph to describe causal relationships nor conducts reasoning over the graph to estimate causal effects.

2.1.3 Causal Effect Estimation and Causal Discovery. For estimating the causal effect, one golden rule is to conduct randomized experiments. Since individuals are divided into the treatment group and the control group randomly, there are no unobserved confounders. Under randomized experiments, some favorable properties of causal inference are guaranteed, such as covariate balance and exchangeability. Meanwhile, the causal effect can be estimated directly by comparing the two groups. For example, online A/B testing can be regarded as a kind of randomized experiment that divides users randomly into several groups and can obtain trustworthy evaluation results of recommendation performance.

However, randomized experiments can be expensive and sometimes impossible to conduct. For example, in recommender systems, experiments generating randomized recommendations can detrimentally affect user experiences and the platform's profitability. Therefore, estimating the causal effect solely from observational data becomes critical. In general, a causal estimand is first transformed into a statistical estimand with a causal model like SCM. Then the statistical estimand is estimated with observed data. In other words, with the defined causal model, we can discern causal effects and non-causal effects, such as confounding associations between treatment and outcome. Subsequently, the causal effect is extrapolated by estimation using observed data in alignment with the identified causal mechanisms.

One classical method is **backdoor adjustment** [71]. We say a set of variables W satisfies *backdoor criterion* if W contain no descendant of T and W can block backdoor paths (which has arrow into T rather than from T) between T and Y. The causal effect of T on Y then can be obtained with backdoor adjustment as follows:

$$P(y \mid \mathrm{do}(t)) = \sum_{w} P(y \mid t, w)P(w), \tag{1}$$

where $w \in W$ and the total causal effect is the weighted sum of the conditioned causal effect.

The above backdoor adjustment can address observed confounders, but not unobserved confounders, where **frontdoor adjustment** [71] comes to help. We say a set of variables M satisfies *frontdoor criterion* if all the causal paths from treatment variable T to the outcome variable Y are through M, and there is no unblocked backdoor path from T to M, as well as M to Y when conditioned on T. The causal effect of T on Y then can be obtained with frontdoor adjustment as follows:

$$P(y \mid \mathrm{do}(t)) = \sum_{m} P(m \mid t) \sum_{t'} P(y \mid m, t')P(t'), \tag{2}$$

where possible unobserved confounders are addressed.

With the sufficient adjustment set of variables W in the high dimension, it is difficult to directly estimate the causal effect as the positivity property is hard to satisfy. Instead of modeling the whole set W, we can turn to the propensity score as follows:

$$e(W) = P(T = 1 \mid W), \tag{3}$$

which indicates the probability of receiving the treatment given W. Then the causal effect can be estimated by **inverse propensity weighting (IPW)** [35] on the treatment and control group as follows:

$$\hat{\tau} = \frac{1}{n_1} \sum_{i:t_i=1} \frac{y_i}{e(w_i)} - \frac{1}{n_2} \sum_{j:t_j=0} \frac{y_j}{1 - e(w_j)}. \tag{4}$$

All of the above causal effect estimations assume that we already have a causal graph. However, in the real world, we often lack prior knowledge about the causal relationships in collected data. This limitation gives rise to the problem of causal discovery, where the objective is to construct a causal graph from the existing data of a set of variables. Traditional approaches identify causal relations through conditional independence tests, bolstered by additional assumptions such as faithfulness [93]. Score-based algorithms [34, 87] have been also proposed to relax the strict assumptions for causal discovery. These methods utilize a score function to measure the quality of the discovered causal graph in comparison with observed data. Recently, various machine learning approaches have been developed to discover causal relations from large-scale data. For example, Zhu et al. [142] utilize reinforcement learning method to find an optimal DAG with respect to a scoring function and penalties on acyclicity. There is a survey [26] fully discusses different methods of causal discovery.

To summarize it, we have introduced the fundamental knowledge of causal inference, including two basic frameworks and two important research topics, causal effect estimation, and causal discovery.

2.2 Recommender System

2.2.1 Overview. As an approach to information filtering, the recommender system has been widely deployed on various platforms in recent decades, such as TikTok, YouTube, X (formerly known as Twitter), and so on. In general, the modeling of user preferences based on historical interactions is the key point for the recommendation algorithm, and users' future interactions are further predicted. In this way, the necessary data input of a recommendation task includes the records of user-item interactions, and the output is a model that can generate the interaction likelihood of a given user-item pair. This procedure can be formulated as

$$\begin{aligned} &\textbf{Input}: \mathbf{Y} \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{I}|}, \\ &\textbf{Output}: f(\cdot,\cdot), (u,i) \xrightarrow{f} \mathbb{R}, \end{aligned} \tag{5}$$

where $\mathcal{U}$ and $\mathcal{I}$ denotes the user set and item set, respectively. $y_{u',i'} = 1$ if user $u' \in \mathcal{U}$ has interacted with item $i' \in \mathcal{I}$; if not, $y_{u',i'} = 0$; here the function $f(\cdot,\cdot)$ denotes the recommendation model. Furthermore, with different input data, there are two primary families of models in recommendation, i.e., CF and CTR prediction. Despite the vanilla CF which only considers user-item interaction data, some recommendation tasks enhance the behavioral data with auxiliary data, such as social network in social recommendation [18, 115], behavioral sequences in sequential recommendation [8, 143], multiple-type behaviors in multi-behavior recommendation [41, 136], multi-domain user behaviors in cross-domain recommendation [20, 37], and so on. For CTR prediction problem, user and item features such as user profiles (occupation, age) and item attributes (category, brand) are also considered as input. The mainstream works of CTR prediction focus on extracting high-order cross-features with attention-based neural network [24], attention-based neural network [24], self-attentive layers [91], and so on.

2.2.2 Recommendation Model Design. Here we present two folds of design of recommendation models, collaborative filtering and click-through rate prediction.

Collaborative Filtering. Following the development process, existing CF models can be categorized into three types, including MF-based, **neural network** (**NN**)-based, and **graph neural network** (**GNN**)-based. The standard way of modeling is to represent users and items with latent vectors, i.e., embeddings. With user embedding matrix $\mathbf{P} \in \mathbb{R}^{d\times|\mathcal{U}|}$ and item embedding matrix $\mathbf{Q} \in \mathbb{R}^{d\times|\mathcal{I}|}$, in which d denotes embedding dimension, the interaction likelihood of (u,i) will be the similarity of corresponding embeddings $\mathbf{p}_u$ and $\mathbf{q}_i$.

— **MF** [47]. The similarity function is the inner product as follows:

$$s(u,i) = \mathbf{p}_u^\top \mathbf{q}_i. \tag{6}$$

— **NCF** [31]. In order to incorporate the capability of modeling non-linearity, NCF generalized the similarity function and introduced the **multi-layer perceptron** (**MLP**) as follows:

$$\begin{aligned} s(u,i) &= \mathbf{h}^\top \left(\mathbf{p}_u^G \odot \mathbf{q}_i^G\right) + \phi\left([\mathbf{p}_u^M, \mathbf{q}_i^M]\right), \\ \mathbf{p}_u &= [\mathbf{p}_u^G, \mathbf{p}_u^M], \mathbf{q}_i = [\mathbf{q}_i^G, \mathbf{q}_i^M], \end{aligned} \tag{7}$$

where $\mathbf{p}_u^G, \mathbf{p}_u^M$ ($\mathbf{q}_i^G, \mathbf{q}_i^M$) denotes the user (item) embedding for MF and MLP parts respectively, $[\cdot,\cdot]$ indicates the concatenation operation, $\odot$ indicates the Hadamard product, $\mathbf{h}$ is the weight vector, and $\phi(\cdot)$ denotes MLP.

— **NGCF** [106]. This GNN-based recommendation model conducts multiple layers of message passing on the user-item bipartite graph. Formally, the similarity is calculated as follows:

$$\begin{aligned} \mathbf{p}_u^l &= \mathbf{Agg}\left(\mathbf{q}_i^{l-1} | i \in \mathcal{N}_u\right), \mathbf{q}_i^l = \mathbf{Agg}\left(\mathbf{p}_u^{l-1} | u \in \mathcal{N}_i\right), \\ s(u,i) &= \phi(\left([\mathbf{p}_u^0, \ldots, \mathbf{p}_u^L]\right)^\top [\mathbf{q}_i^0, \ldots, \mathbf{q}_i^L]), \end{aligned} \tag{8}$$

where $\mathbf{p}_u^0 = \mathbf{p}_u, \mathbf{q}_i^0 = \mathbf{q}_i$, and l indicates the propagation layer, $\mathcal{N}_u$ refers to the set of interacted items of user u, and $\mathcal{N}_i$ indicates the set of those users who have interacted with item i. Here $\mathbf{Agg}(\cdot)$ is the aggregation function for collecting neighborhood information. In this way, high-order user-item connectivity is injected into the similarity measurement between nodes.

Click-Through Rate Prediction. As introduced above, the unified procedure of CTR prediction is extracting high-order features. The input features are denoted as follows:

$$\mathbf{x}_{u,i} = [\mathbf{x}_{u,i}^1, \ldots, \mathbf{x}_{u,i}^M], \tag{9}$$

where M denotes the number of feature fields. Furthermore, the raw features will be transformed into embeddings as follows:

$$\mathbf{v}_{u,i}^k = \mathbf{V}^k \mathbf{x}_{u,i}^k, k = 1, \ldots, M, \tag{10}$$

where $\mathbf{V}^k \in \mathbb{R}^{d^k \times |\mathcal{F}^k|}$ is the feature embedding matrix, $\mathcal{F}^k$ is the set of optional features, d^k is the dimension of embeddings, and k denotes the order of feature field. In general, there are two fields of users' and items' identity, supposed to be the first two ones, then $\mathbf{V}^1 = \mathbf{P}$ and $\mathbf{V}^2 = \mathbf{Q}$. In terms of the mapping function, it can be represented as follows:

$$s(u,i) = g\left([\mathbf{v}_{u,i}^1, \ldots, \mathbf{v}_{u,i}^M]\right). \tag{11}$$

The design of $g(\cdot)$ will introduce a module of feature interaction learning, via the inner product in FM [74], multi-layer perceptions in DeepFM [24], stacked self-attention layers in AutoInt [91], and so on.

2.2.3 Objective Function. The primary objective functions for optimization utilized in recommendation models are in two categories, i.e., point-wise and pair-wise. Specifically, the point-wise objective function focuses on the prediction of a user-item interaction of which the widely-used Logloss function is as follows:

$$\mathcal{L} = -\frac{1}{|O|} \sum_{(u,i) \in O} y_{u,i} \log(\hat{y}_{u,i}) + (1 - y_{u,i}) \log(1 - \hat{y}_{u,i}), \tag{12}$$

where $\hat{y}_{u,i} = s(u,i)$ and O is the training set.

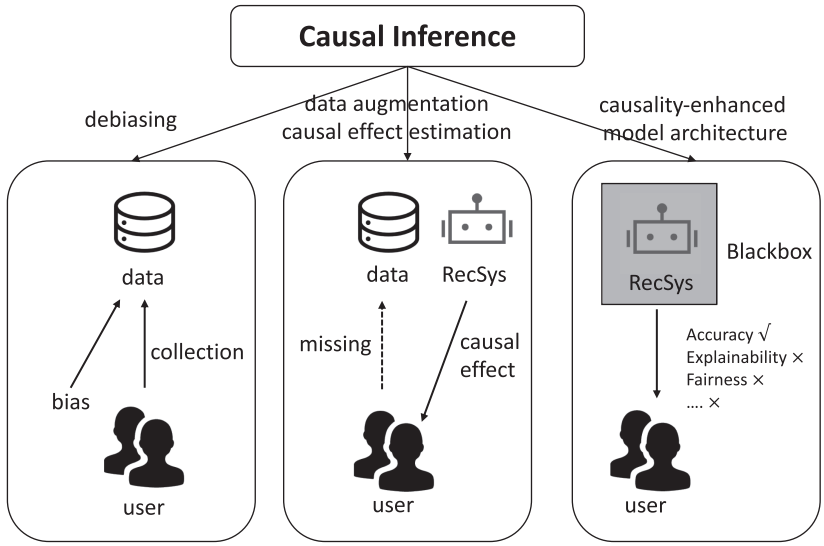

Fig. 4. Illustration of three typical issues of non-causality recommendation models and how causal inference addresses them.

In terms of the pair-wise objective function, it encourages a larger disparity between positive ($y_{u,i} = 1$) and negative ($y_{u,j} = 0$) samples, and the widely-used BPR loss function [75] is as follows:

$$\mathcal{L} = -\frac{1}{|Q|} \sum_{(u,i,j) \in Q, \hat{y}_{u,i} > \hat{y}_{u,j}} \log \sigma(\hat{y}_{u,i} - \hat{y}_{u,j}), \tag{13}$$

where $\sigma(\cdot)$ denotes the sigmoid function, and Q denotes the training set.

3 WHY CAUSAL INFERENCE IS NEEDED FOR RECOMMENDER SYSTEMS

In this section, we will discuss the essentiality and benefits of introducing causal inference into recommender systems from three aspects, illustrated in Figure 4.

3.1 The Issues of Data Bias in Recommender Systems

3.1.1 Data Bias in Recommender Systems. *Data bias* refers to the uneven distribution of recommendation data that does not faithfully reflect user preference. Generally, there are two main types of data bias in recommendation over interactions and attributes.

Bias over interactions. Historical user-item interactions collected from previous recommendation strategies are typically treated as labels for recommender model training. Sometimes, historical interactions follow a highly skewed distribution over items (*a.k.a.* long-tail distribution), resulting in models over-recommend popular items, i.e., **popularity bias** [111, 138]. Furthermore, the historical interactions of a user also exhibit uneven distributions over item categories. Consequently, recommender models will blindly assign high scores to items from the frequent category, ignoring the user preference over the remaining categories [102]. Worse still, such biases will be amplified in the feedback loop, leading to notorious issues like unfairness and the filter bubble. **Conformity bias** refers to the fact that users' behaviors are determined by not only user preferences but also conformity, making the collected data biased. It is a common issue

in social-aware information systems, such as the user-post interaction behavior on Facebook.[1] **Exposure bias** is another widely-concerned bias, which refers to that the exposure algorithms will highly influence the data collection of user feedback.

Bias over attributes. Item attributes that can directly result in interactions, especially clicks, can also mislead the estimation of user preference. Training over historical interactions will inevitably push the model to highlight such attributes, leading to shortcuts. Taking video recommendation as an example, videos with attractive titles or cover images are more likely to be clicked, while the user may not like the content [103]. Undeniably, the shortcuts of such item attributes will lead to recommendations failing to satisfy user preference. Worse still, they also make the recommender system vulnerable to relevant attacks, e.g., the item producer intentionally leverages such features.

3.1.2 The Necessity of Causal Inference for Data Debiasing. Causal theory enables us to identify the root cause of data bias by scrutinizing the generation procedure of recommendation data and mitigating the impact of bias through causal recommendation modeling.

Causal view of data bias. The main source of bias effect in recommendation is the backdoor path (Figure 2(b)), where a confounder (Z) simultaneously affects the inputs (X) and interactions (Y). Due to the existence of the backdoor path, directly estimating the correlation between X and Y will suffer from spurious correlations, leading to a recommendation score higher than X deserved. For instance, item popularity affects the exposure probability of an item in a previous recommendation strategy and interaction probability due to user conformity. Due to ignoring item popularity, CF methods will assign higher scores to items with higher exposure in previous recommendation strategies, leading to over-recommendation, i.e., popularity bias. In the causal terminology, this type of bias effect is termed as *confounding bias*. Beyond confounding bias, another source of bias in recommendation is the gap between the observed interactions and true user preference matching. Some item attributes directly affect the status of interactions.

Causal recommendation modeling. The key to eliminating bias effects lies in modeling the causal effect of X on Y instead of the correlation between them. In causal language, it means viewing X and Y as treatment and outcome variables, respectively. The causal effect denotes to what extent Y changes according to X, i.e., the changes of Y when forcibly changing the value of X from a reference status to the observed value. To estimate such a causal effect, it is thus essential to incorporate conventional causal inference techniques into recommender models. Consider video recommendations on platforms like YouTube and Netflix. Here, X represents user preference while Y symbolizes users' click behaviors. In an ideal setting, users' click behaviors should be directly influenced by their preferences. However, at times, external factors like an enticing video cover (represented by Z) might introduce bias.

3.2 The Issues of Data Missing and Data Noise in Recommender Systems

3.2.1 Data Missing in Recommender Systems. The data utilized in recommender systems is typically limited, which cannot cover all possible user-item feedback. For example, a user has only rated a small ratio of clicked movies; or the user purchasing the camera is not recorded as having bought a camera lens and a roll film, which is intuitively reasonable. Therefore, the obtained data cannot fully represent the users' interest, leading to sub-optimal results for existing recommendation methods. First, the interaction data observed is constrained by the already-deployed recommendation policy of the recommender system [78]. Users can only interact with specific items if these items are exposed to them, which strongly correlates with

[1] https://facebook.com

the recommender system's intrinsic strategy. In addition, users may refuse to give feedback [107]. For example, on movie rating websites such as IMDB or Douban,[2] users may only rate a few of the movies they have watched. Under this condition, it becomes more challenging to model users' interests. Besides, features of users and items can also be missing in real-world recommender systems due to the high cost of feature collection.

3.2.2 The Necessity of Causal Inference for Data Missing. Some earlier approaches [85, 94, 98] without causal inference were developed to address the data-missing problem. Steck [94] computes prediction errors for missing ratings. Schnabel et al. and Thomas et al. [85, 98] consider weights for each observed rating based on the probability of collecting that record. However, these methods are limited by low accuracy and poor generalization ability. Causal inference actually provides the causal descriptions of how data is generated, which can serve as prior knowledge to data-driven models. As a result, the negative impact of data-missing issues can be alleviated, improving accuracy and generalization ability.

3.2.3 Data Noise in Recommender Systems. The recommender systems highly rely on the historical user-item interaction feedback to model users' preferences and predict the interaction probability between the user and the unseen item; thus, the reliability of collected data is the basis of the effectiveness of recommender systems. However, the data collected in the real world may be noisy, i.e., *incorrect*. It is hard to detect and eliminate noisy interactions in traditional recommendation methods. Mahony et al. [65] classified data noise into two categories: *natural noises* and *malicious noises*. Natural noise relates to the noise generated during the data-collection procedure by recommender systems, and malicious noise denotes the noise being deliberately inserted into the system.

As for the natural noise, Li et al. [51] discussed various reasons that lead to the noisy data in recommender systems. The major reasons include the inaccurate impression of the users themself and the error in data collection. Jones et al. [43] points out that users can hardly accurately measure their preferences, thus leading to mismatch between their preferences and final ratings. Cosley et al. [14] found that noisy data arises when users map their opinions into discrete ratings. Zhang et al. [137] argued that in some streaming applications, the conversion events may be delayed to the time when data is collected. Thus the feedback of users may have not yet occurred, resulting in a large number of incompletely labeled instances and introducing noise to data. Some existing work [38, 60, 101, 112] also pointed out the difference between the implicit feedback and users' actual satisfaction because of noisy interactions. For example, in E-Commerce, many clicks do not lead to purchases, and a large portion of purchases finally get negative comments. Implicit interaction data widely used in recommender systems nowadays is easy to become noisy because of the inaccurate first impression of users. Since users are exposed to a flood of information in today's online services, users are very likely to have accidentally triggered feedback such as click-by-mistake.

As for the *malicious noise*, it is produced by adversary attackers of recommender systems. For instance, on user-generated platforms such as TikTok,[3] some authors will create plenty of new accounts to rate their work with high scores, trying to earn over-exposure opportunities. In e-commerce websites such as Amazon, some adversary sellers may generate fake order records or positive comments on their products.

3.2.4 The Necessity of Causal Inference for Data Denoising. Many previous works have experimentally demonstrated the severity of data noise and its negative effects on recommender systems. Cosley et al. [14] showed that only 60% of users will keep their rating to the same movie when they are asked to re-rate for it. Further experiments show that statistically significant

[2] https://www.douban.com
[3] https://www.tiktok.com

MAE differences arise when exploiting CF models on the original rating data and new rating data. Amatriain et al. [3] showed that the recommendation performance will be significantly affected under noisy data compared to the noiseless data, with a difference of RMSE of about 40%. Wang et al. [101] found through experiments on two representative datasets the performance of recommender system trained by noisy data experienced a performance drop of 9.56%–21.81% *w.r.t.* Recall@20 and drop of 3.92%–8.81% *w.r.t.* NDCG@20, compared with the recommender system trained over cleaned data. Although existing work has confirmed the widespread existence of data noise, which reveals that we need to consider its impact during training recommendation models, existing solutions are a few. Data noise can arise from various sources, such as limitations in data collection (e.g., inaccurate values in users' questionnaire data) or during data preprocessing (e.g., crudely transforming feedback into simplified values, like converting continuous user watching durations into discrete positive/negative labels). Such noises present significant challenges in accurately discerning user preferences. Leveraging causal inference allows us to more effectively detect the presence of noise in interaction data or bridge the disparity between noisy training data and the expected clean testing data with the help of counterfactual learning and reasoning.

3.3 Beyond-accuracy Concerns in Recommender Systems

Traditional recommender systems are designed toward the major goal of achieving higher accuracy, i.e., click-through rate or conversion ratio, serving for the platform benefit. Nevertheless, as recommender systems have become fundamental information services in more and more aspects of daily life, these concerns are not just technical problems but also social challenges.

3.3.1 Explainability. The requirement of explainability for recommender systems refers to the need that we should understand why some items are recommended while others are not. It helps build a bridge between users and recommendation lists for better transparency and trustworthiness. Specifically, it can be divided into two categories, explainable recommendation model and explainable recommendation results. Some existing work [11, 100, 139] mainly took some item aspects to give explanations, which is concluded as the aspect-aware explainable recommendation. For example, Wang et al. [100] learned users' preferences on given aspects by factorization method to get the aspect-aware explanations.

The necessity of causal inference. Despite their effectiveness to some extent, existing methods of explainable recommendation are still limited [96]. Specifically, the explanation is built on correlation. As mentioned above, roughly extracting correlations from the observed data without the support of causal inference may lead to wrong conclusions. Furthermore, the explanations of the recommendation model require building explicit causal relations between the components of the recommendation model and the prediction scores. Additionally, the explanation for recommendation results should fully consider how different decision-factors, i.e., cause, lead to users' behaviors, i.e., effect. Thus, achieving explainability is tightly connected to causal inference.

3.3.2 Diversity and Filter Bubble. Filter bubble describes the phenomenon where people tend to be isolated from diverse content and information by online personalization [66]. As a consequence, users are placed in a fixed environment where they can only encounter similar topics or information. Passe et al. [68] attribute this effect to homogenization, which means people's behavior and interest show consistency and convergence.

The recommender system is one of the main causes of the filter bubble due to the principle of generating recommendation lists by learning the similarity between users or items [67], which inevitably leads to homogeneous recommendations. Gabriel Machado Lunardi et al. [61] empirically analyzed the filter-bubble formation based on popular CF methods and algorithms for diversified recommendation. In terms of human nature, researchers found that people tend to

pursue a comfort zone and stay with the opinions they are interested in or agree with [6]. In the long term, the filter bubble will narrow people's views and radicalize their ideas. Thus, it is an urgent problem to break filter bubbles and improve recommendation heterogeneity.

The necessity of causal inference. The biased feedback loop is one of the most critical challenges in addressing the filter bubble, as learning from biased data will exacerbate the homogeneity in recommendation exposure and further bias the collected data. Moreover, the accuracy-diversity dilemma is another challenge, which refers to the phenomenon where pursuing accuracy will lead to low diversity. Causal inference provides the opportunity to address these challenges. First, causal inference can alleviate the bias or missing data in collected data, supporting the exploration of unseen data. Second, the causal inference-enhanced model can utilize the causal relationships under user behaviors, understanding why users consume certain items. This can help recommend items outside the existing categories and meet user demands.

3.3.3 Fairness. Recently, the fairness of recommendations has gained significant attention. As we know, recommender systems operate as multi-stakeholder platforms, thus encompassing various aspects of fairness concerns, including both user-side and item-side [7].

The user-side fairness issue arises from the diverse fairness concerns among users. For instance, while some users may be predominantly worried about potential biases based on their gender, others might be more concerned about age-related biases [54]. To foster trust in the recommender system, it's essential to address these user-side fairness concerns in a personalized manner. While some approaches [28] have attempted to rectify these fairness challenges using association-based methods–aimed at eliminating statistical metric discrepancies between groups–research has shown these methods to be inadequate and lacking in certain areas [45, 48]. Notably, these association-based techniques often overlook the intricate relationship between objective features and model outputs. Conversely, a few studies have explored fairness through a causal lens, offering insights into how output variables evolve with changes in input [1, 44].

Item-side fairness, on the other hand, evaluates the equity in treatment of each item during the recommendation process. Biases may emerge due to the oversight of particular items or their attributes. Several existing solutions [23, 86] have ventured into unbiased learning or heuristic ranking adjustments to rectify these biases.

The necessity of causal inference. Tackling fairness issues is akin to hypothesizing in a counterfactual realm: Had a user not been part of a specific group, or had an item lacked a certain feature, would the recommendation outcomes remain unchanged? If not, what would these altered recommendations look like? This difference between the counterfactual and factual worlds forms the cornerstone of fairness evaluation in recommender systems. Hence, methods grounded in causal inference, particularly those employing counterfactual reasoning, offer a fresh and more comprehensive approach to enhancing recommendation fairness compared to their non-causal counterparts.

In short, we have systematically discussed the limitations of existing recommender systems and why causal inference is essential to address these limitations. In the following, we will introduce how these challenges can be addressed (at least partially addressed) by presenting the recent advances in the causality-enhanced recommendation.

4 TECHNICAL DETAILS OF EXISTING WORKS OF CAUSAL INFERENCE-BASED RECOMMENDER SYSTEMS

The existing work of causal inference for recommendation is presented based on the three major issues of recommendation models with only correlation considered. The overall illustration is presented in Figure 5, and the details are introduced one by one as follows.

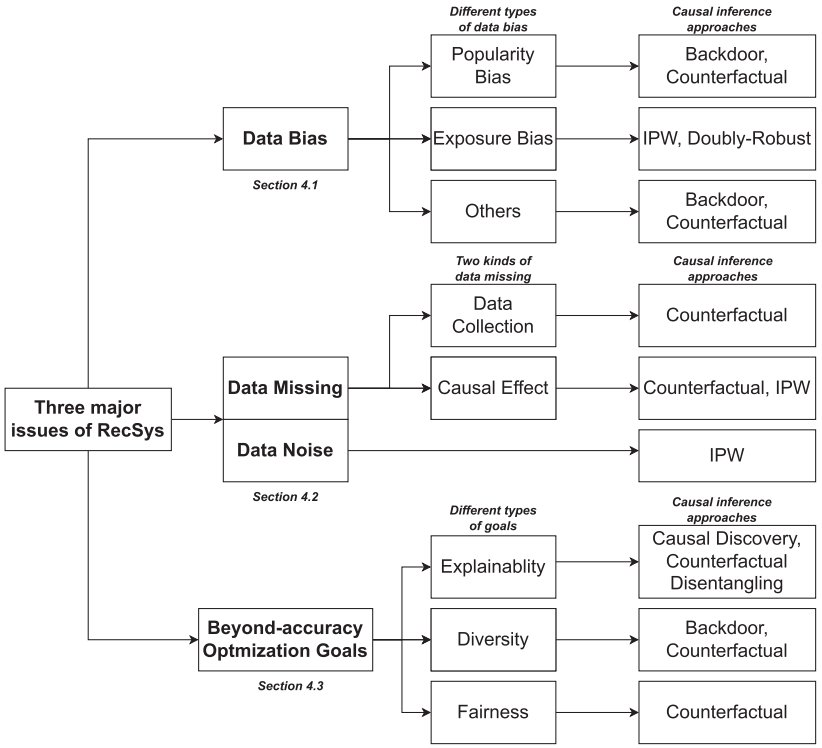

Fig. 5. Illustration of existing work of causal inference for recommendation.

4.1 Causal Inference-based Recommendation for Addressing Data Bias

Existing methods on causal debiasing are mainly in three categories: confounding effect, colliding effect, and counterfactual inference.

4.1.1 Confounding Effect. In most cases, biases are caused by confounders, which lead to confounding effect in correlations estimated from the observations. To eliminate the confounding effect, there are mainly two lines of research regarding the causal inference frameworks adopted.

Structural Causal Model. Using SCM to eliminate confounding effect falls into two categories: backdoor and frontdoor adjustments. Backdoor adjustment is able to remove the correlations by blocking the effect of the observed confounders on the treatment variables. To address the data bias in recommender systems, the existing work usually inspects the causal relationships in the data generation procedure, identifies the confounders, and then utilizes backdoor adjustment to estimate causal effect instead of correlation. Specifically, backdoor adjustment blocks the effect of confounders on the treatment variables by intervention [71], which forcibly adjusts the distribution of treatment variables and cuts off the backdoor path from treatment variables to outcome variables via confounders.

For example, Zhang et al. [138] ascribed popularity bias to the confounding of item popularity, which affects both the item exposure and observed interactions. They then introduced backdoor adjustment to remove the confounding popularity bias during model training, and incorporated an inference strategy to mitigate popularity bias. Besides, Wang et al. [102] explored the bias amplification issue of recommender systems, i.e., over-recommending some majority item categories in users' historical interactions. For instance, recommender systems tend to recommend

Table 1. Representative Methods that Utilize Causal Inference to Address Data Bias

Category	Model	Causal-inference Method	Venue	Year
Popularity Bias	PD [138]	Backdoor Adjustment	SIGIR	2021
	MACR [111]	Counterfactual Inference	KDD	2021
Clickbait Bias	CR [103]	Counterfactual Inference	SIGIR	2021
Bias Amplification	DecRS [102]	Backdoor Adjustment	KDD	2021
Conformity Bias	DICE [141]	Disentangled Causal Embeddings	WWW	2021
Unknown Bias	RD [16]	Doubly-Robust	KDD	2022
General Feature Bias	DCR[32]	Backdoor Adjustment	TOIS	2023
Exposure Bias	IPS [86]	IPW	ICML	2016
	MF-DR-JL [107]	DR	ICML	2019
	Multi-IPW/DR [135]	IPW, DR	WWW	2020
	Rel-MF [81]	IPW	WSDM	2020
	DR [79]	DR	RecSys	2020
	MRDR [27]	DR	SIGIR	2021
	LTD [108]	RCT, DR	SIGIR	2021
	AutoDebias [9]	RCT	SIGIR	2021
	USR [109]	IPW	WWW	2022
	DENC [52]	IPW	TKDD	2023

more action movies to users if they have interacted with a large proportion of action movies before. To tackle this, Wang et al. [102] found that the imbalanced distribution of item categories is actually a confounder, affecting user representation and the interaction probability. Next, the authors proposed an approximation operator for backdoor adjustment, which can help alleviate the bias amplification.

However, the assumption of observed confounders might be infeasible in recommendation scenarios. To tackle the unobserved confounders (e.g., the temperature when users interact with items), frontdoor adjustment is a default choice [71]. Xu et al. [125] has made some initial attempts to address both global and personalized confounders via frontdoor adjustment. Zhu *et al.* [144] gave a more detailed analysis of the conditions to apply the frontdoor adjustment in recommendation. Liu et al. [58] approached the selection bias challenge and proposed counterfactual learning-based method. Specifically, the authors focus on policy learning approaches for top-K recommendations in extensive item spaces, identifying key challenges like importance weight explosion and observation scarcity. A novel framework is introduced for efficient policy learning that addresses these complexities. Ding et al. [16] emphasizes the challenge of unmeasured confounders in recommender systems which can influence the accuracy of feedback predictions. The authors proposed **Robust Deconfounder** (**RD**) to consider the effects of these unmeasured confounders on propensities, using a bounded effect approach.

Potential Outcome Framework. From the perspective of the potential outcome framework, the target is formulated as an unbiased learning objective for estimating a recommender model. Let O^e denote the exposure operation where $o_{u,i} = 1$ means item i is recommended to user u. The set O is defined as the exposure results under the given exposure strategy (with O^e). According to the definition of IPW [56], we can learn a recommender to estimate the causal effect of X on Y by minimizing the following objective,

$$\frac{1}{|O|} \sum_{(u,i) \in O} \frac{l(y_{u,i}, \hat{y}_{u,i})}{\hat{p}_{u,i}}, \tag{14}$$

where $l(\cdot)$ denotes a recommendation loss and $\hat{p}_{u,i}$ denotes the propensity, i.e., the probability of observing the user-item feedback $y_{u,i}$. As one of the initial attempts, Tobias et al. [86] adopted this objective to learn unbiased matrix factorization models where the propensity is estimated by a separately learned propensity model (logistic regression model). Beyond such shallow modeling of propensity [81], Zhang et al. integrated the learning of propensity model and recommendation model into a multi-task learning framework [135], which demonstrates advantages over the separately learned one. Wang et al. [109] took the pioneering step of considering the exposure bias in the sequential recommendation, by proposing an IPW-based method named USR for alleviating the confounder in sequential behaviors.

Nevertheless, estimating the proper propensity score is non-trivial and typically suffers from high variance. To address these issues, a line of research [27, 79, 107] pursues a doubly-robust model estimator by augmenting Equation 14 with an error imputation model, which is formulated as

$$\frac{1}{|\mathcal{U}| \cdot |\mathcal{I}|} \sum_{(u,i)} \left(\hat{e}_{u,i} + \frac{o_{u,i}(l(y_{u,i}, \hat{y}_{u,i}) - \hat{e}_{u,i})}{\hat{p}_{u,i}} \right), \tag{15}$$

where $\hat{e}_{u,i}$ is the output of the imputation model with user-item features as inputs. To learn the parameter of the imputation model, a joint learning framework [107] optimizes:

$$\frac{1}{|O|} \sum_{(u,i) \in O} \frac{(l(y_{u,i}, \hat{y}_{u,i}) - \hat{e}_{u,i})^2}{\hat{p}_{u,i}}. \tag{16}$$

Undoubtedly, incorporating experimental data, i.e., interactions from **randomized controlled trial** (**RCT**) such as random exposure, can enhance the doubly-robust estimator. In this light, a line of research [9, 108] investigates data aggregation strategies, which largely focuses on tackling the sparsity issue of experimental data since RCT is costly. Recently, Li et al. [52] considered that the exposure bias is largely depends on the socially-connected users, and proposed IPS-based methods with the auxiliary social network data.

Different with the existing works for specific kinds of bias, He et al. [32] studied how to address general feature biases. This work identified a challenge in recommender systems where some features, like video length, can bias user interaction data and misrepresent actual preferences. Approaching from a causal perspective, the study introduced the **Deconfounding Causal Recommendation** (**DCR**) framework to address this bias. The DCR used backdoor adjustment to counteract the effects of confounding features and combine it with the **mixture-of-experts** (**MoE**) model architecture.

4.1.2 Colliding Effect. We can discover many collider structures (*cf.* Figure 2(c)) in the interaction generation process by inspecting the causal relationships. A representative case is that many different variables affect the observed interactions, such as user interests and conformity. Conditioning on the collected user interactions will lead to the correlation between user interests and conformity: an interaction caused by user conformity has a higher probability of being uninterested. To mitigate the conformity bias, an existing work [141] disentangles the interest and conformity representations by training over cause-specific data, which improves the robustness and interpretability of user representations.

4.1.3 Counterfactual Inference. Another SCM-based technique used for debiasing is counterfactual inference. In some SCMs of recommender systems, two causes (user and item features) lead to one effect (user behavior). If the user features or item features are significantly biased, this direct path (which we inaccurately referred to as a "shortcut' in our original version) can result in biased interaction learning, especially when other unbiased features take a more indirect route. The

Table 2. Representative Methods that Utilize Causal Inference to Address Data Missing and Data Noise (RecSys Refers to Recommender System and CI Refers to Causal Inference)

Category	Model	RecSys Task	CI Method	Venue	Year
Data Missing	ULO [83]	Collaborative Filtering	Uplift, IPW	RecSys	2019
	DLCE [84]	Collaborative Filtering	IPW	RecSys	2020
	CauseRec [133]	Sequential	Counterfactual	SIGIR	2021
	CASR [110]	Sequential	Counterfactual	SIGIR	2021
	CF^2 [122]	Feature-based	Counterfactual	CIKM	2021
	CPR [127]	Collaborative Filtering	SCM, Counterfactual	CIKM	2021
	CBI [82]	Collaborative Filtering	Interleaving, IPW	RecSys	2021
	CausCF [121]	Collaborative Filtering	Uplift, RDD	CIKM	2021
	DRIB [120]	Collaborative Filtering	Doubly-Robust, IPW	WSDM	2022
	COR [105]	CTR	Counterfactual	WWW	2022
	CausPref [33]	Collaborative Filtering	Causal Discovery	WWW	2022
	ASCKG-CG [64]	KG-based	Counterfactual	SIGIR	2022
	CIRS [22]	Sequential Recommendation	Counterfactual	TOIS	2023
Data Noise	CBDF [137]	Streaming	Importance Sampling	SIGIR	2021

counterfactual inference is able to estimate the path-specific causal effect and eliminate the causal effect of partial user/item features. Specifically, it first imagines a counterfactual world without these features along specific paths and then compares the factual and counterfactual worlds to estimate the path-specific causal effect. For example, Wang et al. [103] conducted counterfactual inference to remove the effect of exposure features (e.g., attractive titles) for mitigating clickbait issues. In addition, Wei et al. [111] reduced the direct causal effect from the item node to the ranking score to alleviate popularity bias. Furthermore, Xu et al. [123] proposed an adversarial component to capture the counterfactual exposure mechanism and optimized the candidate model over the worst-case scenario with a min-max game between two recommendation models.

4.2 Causal Inference-based Recommendation for Addressing Data Missing and Noise

Data collected from recommender systems are usually scarce due to limited user engagement compared with the whole item candidate pool. In addition, the data can also be unreliable and incorrect since the system may fail to collect the true reward within the tight time window for data collection. Meanwhile, the real causal effect of recommendation is largely unknown since the data of *not recommending an item* is unavailable. As a consequence, it is challenging for recommender systems to capture user preferences accurately since they are trained with missing and noisy data. Tools of causal inference can be leveraged to tackle the two problems by generating either counterfactual data to augment insufficient training samples or counterfactual rewards to adjust noisy data. Uplift modeling is utilized to measure the causal effect of recommendation. Table 2 provides a brief summary of recommender systems that utilize causal inference to address data missing and data noise problems.

4.2.1 Causal Inference for Data Missing. Interactions between users and items are the **factual** data, which expresses what really happens on the recommendation platforms and directly reflects user interest. However, factual data is usually scarce; thus, it is insufficient for recommender systems to accurately capture the user interest hidden in the data. The natural idea is to generate more samples that did not actually happen to augment the training data. Such data augmentation aims to answer a question in **counterfactual** world: "what would ... if ...", which has been adopted in

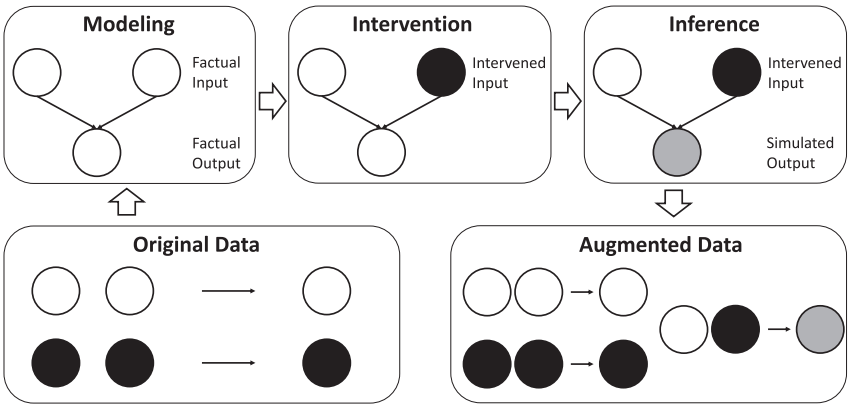

Fig. 6. Illustration of counterfactual data augmentation for data missing.

several research fields like computer vision [19], and natural language processing [145]. In terms of recommendation, counterfactual data augmentation aims to generate more interactions under situations that are different from the real cases when the factual data is collected.

Existing approaches answer counterfactual questions for the following recommendation scenarios,

— **Collaborative Filtering (Top-N Recommendation).** In this scenario, users are provided with a ranked list of items, and they will interact with several items in the list. Data augmentation generates the feedback of unseen recommendation lists; thus the counterfactual question is "what would the given user's feedback be if the system had provided a different recommendation list?" [127].

— **Sequential Recommendation.** In this scenario, recommendation is made according to the historical interaction sequences of users. In other words, interactions of the same user are regarded as a sequence ordered by the timestamp of each interaction. Augmented data are interaction sequences that do not exist in the real scenario. Therefore, the counterfactual question is "what would users behave if their interaction sequences were different?"[110, 133].

— **Feature-based Recommendation.** In this scenario, not only interactions but also features such as user profiles and item attributes are available for recommendation. In other words, user preference modeling can rely on the user/item features. The counterfactual question that data augmentation aims to answer is "what would the given user's feedback be if his/her feature-level preference had been different?" [122].

For all the above three scenarios, counterfactual data augmentation follows a similar paradigm of three steps, modeling, intervention, and inference. Figure 6 provides a brief illustration of counterfactual data augmentation, and we will now introduce these three steps separately.

The **modeling** step captures the data generation process, which can be the recommendation model itself or another separate model. Specifically, it is usually a parametric model that is trained to fit factual data. In other words, given specific users and items that exist in factual data, the model serves as a **simulator** that is trained with the observed interactions and later generates unobserved interactions. For example, Yang et al. [127] first constructs a structural causal model to express the process of recommendation and then implements the SCM with an inner product between user and item embeddings. Xiong et al. [122] utilizes a multi-layer neural network that takes feature vectors of users and items as input, then uses merging operators such as element-wise

product or attention to fuse user and item feature-level properties. Zhang et al. [133] and Wang et al. [110] propose model-agnostic counterfactual data augmentation thus the model can be off-the-shelf sequential recommendation models. The simulator is trained with existing factual data just as a normal recommendation task. After a well-trained simulator is obtained, the input is intervened to be different from factual cases, and the simulator is used to produce the counterfactual outcome. Gao [22] studied **counterfactual interactive recommender system** (**CIRS**), which combines offline reinforcement learning with causal inference. The authors used a causal user model derived from historical data to understand the overexposure effect on user satisfaction, with which model, the RL policy can be better planned.

In the **intervention** step, the input is set as different values from the factual data. Specifically, this step generates the counterfactual cases either by heuristic or another learning-based model. Heuristic-based counterfactual intervention is usually achieved by randomization. In [110], a counterfactual interaction sequence can be generated by replacing an item at a random index with a random item. In [133], dispensable and indispensable items are replaced with random items to construct counterfactual positive and negative sequences, respectively. In contrast, the learning-based counterfactual intervention aims to construct more informative samples as data augmentation. In other words, it generates counterfactual data with higher importance for model optimization. For example, in [127], a counterfactual recommendation list is generated by selecting items with larger loss value i.e., the hard samples. In [110] and [122], items and feature-level preferences that are at the decision boundary are selected and then modified with minimal change to construct more effective counterfactual interaction sequences and input features, respectively.

In the **inference** step, counterfactual outputs are generated with the above counterfactual inputs and simulator. This step which uses the simulator to simulate the output of the intervened input, is usually straightforward. In [127], the counterfactual *clicked* items of the intervened recommendation list are generated by inferring according to the constructed SCM. In [133] and [110], the intervened interaction sequences are fed into the sequential backbone model, and the obtained outputs can directly serve as the counterfactual user embeddings [133], or they can be used to derive counterfactual next items [110].

Wang et al. [105] further considered the problem of out-of-distribution recommendation, i.e., the data in another distribution is missing. The authors proposed to use a variational auto-encoder to help learn the user representations in the counterfactual distribution. Mu et al. [64] proposed to use counterfactual generator to obtain user-item interaction data with the item's specific relation on the knowledge graph is changed. The counterfactual generator and recommender can be trained jointly to enhance each other.

A recent work [33] approaches the issue of data-missing from another perspective, causal discovery. Specifically, this work delves into the vulnerability of current recommender systems to distribution shifts (the missing of IID data). The authors propose a novel causal preference-based recommendation framework named CausPref, integrating a recommendation-specific DAG learner. With emphasizing causal learning of invariant user preference and anti-preference negative sampling, CausPref shows superiority and interpretability in varied OOD settings.

4.2.2 Causal Inference for Data Noise. Interactions can be noisy or incorrect due to the tight time window of data collection. For example, users' feedback can be delayed after the immediate interaction, such as purchasing an item a few days after adding it to the shopping cart. In real-time recommendation, these samples are used for model training before the complete reward is observed. Therefore, the reward at an early time is noisy, and whether the item will be purchased is unknown when it is added to the shopping cart. Zhang et al. [137] tackle the above problem of delayed feedback with the help of causal inference. Specifically, the authors utilize importance

sampling [5, 130] to re-weight the original reward and obtain the modified reward in counterfactual world.

In addition, noisy user feedback can be alleviated by incorporating reliable feedback (e.g., ratings). However, reliable feedback is usually sparse, leading to insufficient training samples. To solve the sparsity issue, Wang et al. [101] contributed a colliding inference strategy, which leverages the colliding effect [71] of reliable feedback on the predictions to facilitate the users with sparse reliable feedback.

4.2.3 Causal Effect Estimation for Recommendation. The recommender systems impact data collection, resulting in the absence of real interaction data, as mentioned above. Existing recommendation approaches are primarily evaluated and trained using interaction data, where typically, more interactions with recommended items indicate a more successful recommendation. However, they overlook the fact that some items may be interacted with by users even without a recommendation. Take e-commerce recommendation as an example: users may have clear intentions and directly purchase the items they desire. On the contrary, some items are more effective in terms of recommendation, meaning that users will purchase these items if recommended but won't purchase them if not recommended. Consequently, recommender systems boost the purchase probability of these effective items, referred to as *uplift*. These items reflect a stronger causal effect of recommendation, emphasizing the importance of recommending more items with a larger uplift.

Some studies [82–84, 121] in recent years investigated the causal effect of recommender systems from the perspective of uplift. Sato et al. [83] applied the potential outcome framework to obtain the **average treatment effect** (**ATE**) of recommendation. Specifically, all the interactions are divided into four categories according to the treatment (recommendation) and the effect (user feedback), and then a sampling approach named ULO is proposed to learn the uplift of each sample. IPW was adopted to achieve unbiased offline learning [84] and online evaluation [82] on the causal effect estimation of recommendation. Xie et al. [121] proposed to estimate the uplift with tensor factorization by regarding treatment as an extra embedding, and they use **regression discontinuity design** (**RDD**) analysis to simulate randomized experiments. Xiao et al. [120] proposed a doubly-robust estimator, along with which a deep variational information bottleneck method is proposed to aid the adjustment of causal effect estimation.

Other studies view the causal effect of the recommendation algorithm as a problem related to off-policy evaluation. In reinforcement learning, the policy determines how the agent behaves (i.e., selecting the action) given the environmental context and the current states. In response, the environment provides the corresponding reward [50]. However, due to high costs and limitations in data collection, it is challenging to collect all possible rewards for every action. Consequently, researchers have proposed off-policy evaluation, aiming to estimate these rewards [2, 77]. In the context of recommendation, the items recommended can be viewed as the policy, and the off-policy evaluation is understood as estimating the effect of the deployed algorithm. This is akin to uplift modeling but focuses more on a general framework that estimates rewards using historical data. To achieve off-policy evaluation, there are three major categories of estimators: model-based estimators (reward regression), model-free estimators like propensity score-based methods, and hybrid estimators using doubly robust methods [80]. Specifically, Swaminathan et al. [95] tackled the problem of slate recommendation, where an ordered set of items is recommended. They built on techniques from combinatorial bandits to estimate a policy's performance using logged data. Li et al. [53] addressed a similar issue, aiming to estimate the number of clicks for a recommendation list. They introduced click models to construct estimators that learn with statistical efficiency, and the results showed the superior performance of these constructed estimators. Mcinerney et al. [62] studied sequential recommendation and introduced a new counterfactual estimator that

Table 3. Representative Methods that Utilize Causal Inference to Achieve Beyond-Accuracy Objectives (RecSys Refers to Recommender System and CI Refers to Causal Inference)

Category	Model	RecSys Task	CI Method	Venue	Year
Explanability	PGPR [118]	KG-enhanced	Causal Discovery	SIGIR	2019
	CountER [97]	CF	Counterfactual & Causal Discovery	CIKM	2021
	MCT [99]	CTR	Couterfactual	KDD	2021
	CLSR [140]	Sequential	Disentangled Embedding	WWW	2022
	IV4Rec [90]	CTR	Decomposed Embeddings	WWW	2022
Diversity	DecRS [102]	CF	Backdoor Adjustment	KDD	2021
	UCRS [104]	CTR	Counterfactual	SIGIR	2022
	∂CCF [124]	CF	Backdoor Adjustment	CIKM	2022
Fairness	CBDF [137]	CTR	Counterfactual	SIGIR	2021

accounts for sequential interactions in the rewards, achieving lower variance. Specifically, they reweighted the rewards in the logging policy to approximate the expected sum of rewards under the target policy. Kiyohara et al. [46] based their work on the assumption that users interact with items sequentially, starting from the top position in a ranking, leading them to propose a Cascade Doubly Robust estimator.

4.3 Beyond-accuracy RecSys with Causal Inference

As mentioned in Section 3.3, non-causal recommender systems may find themselves focusing solely on improving accuracy, potentially overlooking other critical objectives such as explainability, fairness, diversity, and more. In this section, we elaborate on how existing work addresses this challenge by introducing causal inference into recommender systems.

4.3.1 Causal Inference for Explainable Recommendation. Causal inference naturally can improve the explainability of recommendation, since it captures how different factors (cause) leads to recommendation (effect) rather than only the correlations. To present the existing works, we divide them into three categories as follows.

— *Counterfactual learning.* Tan et al. [97] proposed CountER for explainable recommendation using counterfactual reasoning. CountER explained the recommendation by highlighting the distinctions between factual and counterfactual scenarios. Specifically, CountER included an optimization task with the goal of identifying an item that minimizes the difference to the original item, thereby reversing the recommendation outcome in the counterfactual world. CountER [97] also used causal discovery techniques to extract causal relations from historical interactions and the recommended items to enhance the explanation.

— *Causal graph-guided representation learning.* Zheng et al. [140] built a recommendation model based on the causal graph. The authors pre-define the causal relationships that how user behaviors (effect) are generated from users' two parts of preferences (causes), long-term preferences and short-term ones. Long-term preferences refer to those stable and intrinsic interests, while short-term preferences refer to dynamic and temporary interests. The evolution manner is also defined for these two kinds of preferences. Based on the pre-defined causal relations, the authors proposed to assign two disentangled embeddings for two parts of preferences, and the extracted self-supervised signals make the recommendation model explainable. Si et al. [90] proposed to improve the recommendation model's explainability by decomposing model parameters into two parts: causal part and non-causal part. Specifically,

it built a model-agnostic framework by using users' search behaviors as an instrumental variable.

— *Causal discovery.* Xian et al. [118] proposed to make use of a knowledge graph for explainable recommendation, and the paths in the knowledge graph can be used for generating explanations. For example, the reason for purchasing AirPods may be that the user has purchased an iPhone before, and iPhone, and AirPods are reachable in the knowledge graph via relation *has_brand* and node *Apple Brand*. Based on the knowledge graph and users' interaction history, the authors [118] proposed to extract causal relations by a reinforcement learning method. Specifically, the policy function of reinforcement learning is optimized to explicitly select items via paths in knowledge graph, ensuring high performance of both accuracy and explanation. Tran et al. [99] approached the problem of explainable job-skill recommendation. Specifically, it is essential to know which skill to learn to meet the requirements of the job. The authors first proposed causal-discovery methods based on different features with the employment-status label. Then the authors proposed a counterfactual reasoning method that finds the most important feature, of which the modification can lead to employment, which served as the explanations.

4.3.2 Causal Inference for Improving Diversity and Alleviating Filter Bubble. As mentioned earlier, focusing solely on accuracy gives rise to the issue of overly homogeneous content, resulting in the phenomenon known as the filter bubble. By leveraging causal inference, which aids in gaining a deeper understanding and explicitly modeling the causal effects of user-decision factors, recommendations with improved diversity and the reduction of the filter bubble can be achieved.

— *Counterfactual learning.* Wang et al. [104] proposed a causal inference framework to alleviate the filter bubble with the help of user control. Specifically, the framework allows users' active control commands with different granularity to seek out-of-bubble contents. Furthermore, the authors proposed a counterfactual learning method that generates new user embeddings in the counterfactual world to remove user representations of out-of-date features. By constructing counterfactual representations, the recommendation can keep both accurate and diverse.

— *Backdoor Adjustment.* Wang et al. [102] approached the problem of homogeneous recommendation, by regarding imbalanced item distribution as a confounder between user embedding and the prediction score. Specifically, the authors used the backdoor adjustment to block the effect of the imbalanced item-category distribution in training data, partly alleviating filter bubble. The proposed method is model agnostic and thus it can be adapted to different recommendation models, including both collaborative filtering and click-through rate prediction. Xu et al. [124] employed a causal graph with loops to represent the dynamic recommendation process which leads to the filter bubble. A **Dynamic Causal Collaborative Filtering** (**∂CCF**) model is proposed, which leverages back-door adjustment to estimate post-intervention user preferences and employs counterfactual reasoning to alleviate the echo chamber effect. Real-world dataset experiments validate the efficacy of the model in mitigating echo chambers, while maintaining strong recommendation performance.

4.3.3 Causal Inference for Fairness in Recommendation. The concept of achieving fairness naturally aligns with the counterfactual world in causal inference. For instance, when evaluating the fairness of a recommender system for a specific user profile, one can pose a counterfactual question: *Would the recommendation results change if the user profile were altered?* Li et al. [54] introduced the notion of counterfactual fairness in recommendation, where modifying the value of a given feature ensures that the distribution of recommendation probabilities remains unchanged. The

authors address this issue by introducing personalized fairness criteria for users. The core idea is to acquire user embeddings that are independent of specific features. To accomplish this, they propose a filtering module positioned after the embedding layer, which eliminates information relevant to sensitive features and generates filtered embeddings. Subsequently, the authors introduce a prediction module that utilizes these filtered embeddings to predict sensitive features, employing an adversarial learning approach in conjunction with the primary recommendation loss functions.

5 OPEN PROBLEMS AND FUTURE DIRECTIONS

We discuss important yet not-well-explored research directions in causal inference-based recommender systems.

5.1 Causal Discovery for Recommendation

We have systematically reviewed numerous works that integrate causal inference into recommender systems. However, existing approaches relying on predefined causal graphs or structural causal models exhibit two significant limitations.

First, the assumed causal relationships may be inaccurate. Although the recommendation tailored to the causal relations may improve the recommendation performance, hidden variables may exist that are the real causes. Second, these manually crafted causal graphs are often simplistic, typically involving only a few variables, such as the user conformity, user interest, and user behavior in DICE [141], the exposure feature, user/item/context features, and prediction score in CR [110]. Nevertheless, users' decision-making processes may involve many factors in real-world scenarios. For example, whether a user visits a restaurant depends on the location, cuisine, brand, price, and so on. Therefore, it is essential to design causal discovery methods for learning causal relations from real-world data in recommender systems. Traditional methods for causal discovery can be categorized into the following types. **Constraint-based** (**CB**) algorithms, such as the PC algorithm [93] and the FCI algorithm [92], initially identify conditional independence relationships between pairs of variables and then construct a directed acyclic graph based on these relationships. GES methods [13, 73] extend CB algorithms by incorporating a scoring function to assess the suitability of a **directed acyclic graph** (**DAG**). However, these established methods still grapple with challenges like high computational costs and limited robustness when dealing with large-scale data [26]. Recently, novel approaches based on deep learning [42, 59, 88] and reinforcement learning [142] have emerged to infer causal relationships from extensive datasets. Therefore, it is a promising and crucial future direction for discovering causal relations and then leveraging the learned causal relations to enhance recommendation.

5.2 Causality-aware Stable and Robust Recommendation

Recommender systems are expected to be highly stable and robust, which can be explained in the following aspects. First, the utilized data is dynamically collected, such as newly-registered users, new products, and so on. As a result, the data distribution may be fast-changing [105]. Secondly, there exist multiple recommendation scenarios, including different tabs within the same mobile app, diverse domains, and various objectives. This necessitates that the recommendation model be capable of maintaining robustness and stability across these scenarios. Last, there exists a disparity between offline evaluations and online experiments. A recommendation model that performs well in offline experiments should ideally deliver strong results in online environments. In pursuit of greater stability and robustness in machine learning models, prior research [40, 57] has underscored the potential of causality-aware models. These models demonstrate a promising ability to adapt to different domains and excel in **out-of-distribution** (**OOD**) generalization [105].

Therefore, harnessing causality for the enhancement of robust and stable recommendations holds significant importance.

5.3 Causality-aware Graph Neural Network-based Recommendation

In recent years, graph neural networks have been developing in recommendation at an unexpectedly fast speed. GNN-based models have achieved strong performance in various recommendation tasks, such as the significant performance improvement of LightGCN [30] against traditional neural network models [31] in collaborative filtering tasks. The success of graph neural networks is mainly due to the strong ability to extract structured information, especially for the high-order similarity on the graph. However, several critical challenges remain, awaiting solutions bolstered by causality. First, there is a pressing need to demystify the workings of GNNs in making precise and successful recommendations. The explainability of powerful GNN-based recommendation models, encompassing both the model itself and the rationale behind recommendation results, remains an area ripe for further research. Currently, these models often operate as black boxes. Second, while recent strides have been made in causality-aware recommendation models that incorporate GNN modules as integral components, the GNN module itself and the realm of causal inference remain somewhat separate. Explicitly intertwining the message-passing processes of GNNs with causal inference and reasoning for recommendation represents an open and uncharted research frontier.

5.4 Causality-aware Simulator and Environment for Recommendation

The recommender system is a kind of system that tries to estimate and recover how humans make decisions. With a longer-term and more rational objective, such systems should not merely predict current or next-step user interactions but also take into account sequences of interactions, with the aim of maximizing user engagement or aligning with platform requirements. Given the dynamic nature of user-system interactions, some prior works [39, 89] have introduced simulators for recommender systems. These works specifically employ reinforcement learning techniques, including imitation learning [36], to simulate how users select items within specific environments and contexts. However, these approaches are predominantly data-driven and often lack the underpinning of causality, potentially leading to inaccuracies in decision-making processes. Recently, **causal reinforcement learning** (**CRL**) methods have emerged to address the issue of missing data in reinforcement learning tasks. Bareinboim et al.[4] introduced the concept of leveraging causal interventions to aid in estimating rewards while accounting for unobserved confounders. Additional works [49, 126] have delved into causal bandit algorithms, offering theoretical bounds on performance improvements compared to non-causal bandits. Causally-aware reinforcement learning approaches exhibit substantial promise in handling data limitations when modeling dynamic and sequential user-system interactions. Consequently, they are poised to play an indispensable role in modeling both the simulator and the environment of recommender systems.

To conclude, the future endeavors in the realm of causality-aware recommender systems should begin by addressing the constraints imposed by pre-defined causal graphs. Other promising avenues of research encompass enhancing robustness, which involves domain generalization, devising improved evaluation methods for long-term utility, bridging the gap between offline and online settings, exploring more effective integration with graph neural networks, and the development of causality-supported simulators for recommender systems.

6 CONCLUSION

In recent years, causal inference has emerged as a critically significant and transformative topic within the realm of recommender systems research. Its significance cannot be overstated, as it has fundamentally altered our understanding of recommendation models. This article represents an

initial stride toward presenting a comprehensive survey of existing literature in this domain. It meticulously and systematically delves into the rationale behind the applicability of causal inference and how it effectively mitigates the shortcomings inherent in non-causal recommendation models. Our primary aim is to serve as a source of motivation for researchers already active in this field and, equally importantly, to inspire those who are contemplating the initiation of research endeavors in this exciting and burgeoning area.

REFERENCES

[1] Junzhe Zhang and Elias Bareinboim. 2018. Fairness in decision-making – the causal explanation formula. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[2] Aman Agarwal, Soumya Basu, Tobias Schnabel, and Thorsten Joachims. 2017. Effective evaluation using logged bandit feedback from multiple loggers. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 687–696.

[3] Xavier Amatriain, Josep M. Pujol, and Nuria Oliver. 2009. I like it... i like it not: Evaluating user ratings noise in recommender systems. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*. Springer, 247–258.

[4] Elias Bareinboim, Andrew Forney, and Judea Pearl. 2015. Bandits with unobserved confounders: A causal approach. *Advances in Neural Information Processing Systems* 28 (2015), 1342–1350.

[5] Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X. Charles, D. Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. 2013. Counterfactual reasoning and learning systems: The example of computational advertising. *Journal of Machine Learning Research* 14, 11 (2013).

[6] Engin Bozdag, Qi Gao, Geert Jan Houben, and Martijn Warnier. 2014. Does offline political segregation affect the filter bubble? an empirical analysis of information diversity for dutch and turkish twitter users. *Computers in Human Behavior* 41, C (2014), 405–415.

[7] Robin Burke. 2017. Multisided fairness for recommendation. arXiv:1707.00093. Retrieved from https://arxiv.org/abs/1707.00093

[8] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential recommendation with graph neural networks. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 378–387.

[9] 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 *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 21–30.

[10] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.

[11] Tong Chen, Hongzhi Yin, Guanhua Ye, Zi Huang, Yang Wang, and Ming-Chieh Wang. 2020. Try this instead: Personalized and interpretable substitute recommendation. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[12] Yongjun Chen, Zhiwei Liu, Jia Li, Julian McAuley, and Caiming Xiong. 2022. Intent contrastive learning for sequential recommendation. In *Proceedings of the ACM Web Conference 2022*. 2172–2182.

[13] David Maxwell Chickering. 2002. Optimal structure identification with greedy search. *Journal of Machine Learning Research* 3, Nov (2002), 507–554.

[14] Dan Cosley, Shyong K. Lam, Istvan Albert, Joseph A. Konstan, and John Riedl. 2003. Is seeing believing? how recommender system interfaces affect users' opinions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 585–592.

[15] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. 191–198.

[16] Sihao Ding, Peng Wu, Fuli Feng, Yitong Wang, Xiangnan He, Yong Liao, and Yongdong Zhang. 2022. Addressing unmeasured confounder for recommendation with sensitivity analysis. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 305–315.

[17] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *Proceedings of the World Wide Web Conference*. 417–426.

[18] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *Proceedings of theWorld Wide Web Conference*. 417–426.

[19] Tsu-Jui Fu, Xin Eric Wang, Matthew F. Peterson, Scott T. Grafton, Miguel P. Eckstein, and William Yang Wang. 2020. Counterfactual vision-and-language navigation via adversarial path sampler. In *Proceedings of theEuropean Conference on Computer Vision*. Springer, 71–86.

[20] Chen Gao, Xiangning Chen, Fuli Feng, Kai Zhao, Xiangnan He, Yong Li, and Depeng Jin. 2019. Cross-domain recommendation without sharing user-relevant data. In *Proceedings of the World Wide Web Conference*. 491–502.
[21] Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, and Depeng Jin. 2019. Neural multi-task recommendation from multi-behavior data. In *Proceedings of the 2019 IEEE 35th International Conference on Data Engineering*. IEEE, 1554–1557.
[22] Chongming Gao, Shiqi Wang, Shijun Li, Jiawei Chen, Xiangnan He, Wenqiang Lei, Biao Li, Yuan Zhang, and Peng Jiang. 2023. CIRS: Bursting filter bubbles by counterfactual interactive recommender system. *ACM Transactions on Information Systems* 42, 1 (2023), 1–27.
[23] Alois Gruson, Praveen Chandar, Christophe Charbuillet, James McInerney, Samantha Hansen, Damien Tardieu, and Ben Carterette. 2019. Offline evaluation to make decisions about playlistrecommendation algorithms. *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*.
[24] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 1725–1731.
[25] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering* (2020).
[26] Ruocheng Guo, Lu Cheng, Jundong Li, P. Richard Hahn, and Huan Liu. 2020. A survey of learning causality with data: Problems and methods. *ACM Computing Surveys* 53, 4 (2020), 1–37.
[27] Siyuan Guo, Lixin Zou, Yiding Liu, Wenwen Ye, Suqi Cheng, Shuaiqiang Wang, Hechang Chen, Dawei Yin, and Yi Chang. 2021. Enhanced doubly robust learning for debiasing post-click conversion rate estimation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 275–284.
[28] Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of opportunity in supervised learning. In *Proceedings of the Advances in Neural Information Processing Systems*.
[29] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 639–648.
[30] Xiangnan He, Kuan Deng, Xiang Wang, Yaliang Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and powering graph convolution network for recommendation. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*.
[31] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web*. 173–182.
[32] Xiangnan He, Yang Zhang, Fuli Feng, Chonggang Song, Lingling Yi, Guohui Ling, and Yongdong Zhang. 2023. Addressing confounding feature issue for causal recommendation. *ACM Transactions on Information Systems* 41, 3 (2023), 1–23.
[33] Yue He, Zimu Wang, Peng Cui, Hao Zou, Yafeng Zhang, Qiang Cui, and Yong Jiang. 2022. Causpref: Causal preference learning for out-of-distribution recommendation. In *Proceedings of the ACM Web Conference 2022*. 410–421.
[34] David Heckerman, Dan Geiger, and David M. Chickering. 1995. Learning bayesian networks: The combination of knowledge and statistical data. *Machine Learning* 20, 3 (1995), 197–243.
[35] Keisuke Hirano, Guido W. Imbens, and Geert Ridder. 2003. Efficient estimation of average treatment effects using the estimated propensity score. *Econometrica* 71, 4 (2003), 1161–1189.
[36] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. *Advances in Neural Information Processing Systems* 29 (2016).
[37] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 667–676.
[38] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 2008 8th IEEE International Conference on Data Mining*. 263–272.
[39] Eugene Ie, Chih-wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui Wu, and Craig Boutilier. 2019. Recsim: A configurable simulation platform for recommender systems. arXiv:1909.04847. Retrieved from https://arxiv.org/abs/1909.04847
[40] Dominik Janzing. 2019. Causal regularization. *Advances in Neural Information Processing Systems* 32 (2019), 12704–12714.
[41] Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. 2020. Multi-behavior recommendation with graph convolutional networks. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 659–668.
[42] Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *Proceedings of the International Conference on Machine Learning*. PMLR, 3020–3029.

[43] Nicolas Jones, Armelle Brun, and Anne Boyer. 2011. Comparisons instead of ratings: Towards more stable preferences. In *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*. IEEE, 451–456.

[44] Junzhe Zhang and Elias Bareinboim. 2018. Equality of opportunity in classification: A causal approach. In *Proceedings of the Advances in Neural Information Processing Systems*.

[45] Aria Khademi, Sanghack Lee, David Foley, and Vasant G. Honavar. 2019. Fairness in algorithmic decision making: An excursion through the lens of causality. *The World Wide Web Conference* (2019).

[46] Haruka Kiyohara, Yuta Saito, Tatsuya Matsuhiro, Yusuke Narita, Nobuyuki Shimizu, and Yasuo Yamamoto. 2022. Doubly robust off-policy evaluation for ranking policies under the cascade behavior model. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*. 487–497.

[47] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).

[48] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual fairness. In *Proceedings of the Advances in Neural Information Processing Systems*.

[49] Finnian Lattimore, Tor Lattimore, and Mark D. Reid. 2016. Causal bandits: Learning good interventions via causal inference. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*. 1189–1197.

[50] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. arXiv:2005.01643. Retrieved from https://arxiv.org/abs/2005.01643

[51] Dongsheng Li, Chao Chen, Zhilin Gong, Tun Lu, Stephen M. Chu, and Ning Gu. 2019. Collaborative filtering with noisy ratings. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 747–755.

[52] Qian Li, Xiangmeng Wang, Zhichao Wang, and Guandong Xu. 2023. Be causal: De-biasing social network confounding in recommendation. *ACM Transactions on Knowledge Discovery from Data* 17, 1 (2023), 1–23.

[53] Shuai Li, Yasin Abbasi-Yadkori, Branislav Kveton, Shan Muthukrishnan, Vishwa Vinay, and Zheng Wen. 2018. Offline evaluation of ranking policies with click models. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1685–1694.

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

[55] Chen Lin, Xinyi Liu, Guipeng Xv, and Hui Li. 2021. Mitigating sentiment bias for recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 31–40.

[56] Roderick J. A. Little and Donald B. Rubin. 2019. *Statistical Analysis with Missing Data*. John Wiley & Sons.

[57] Chang Liu, Xinwei Sun, Jindong Wang, Haoyue Tang, Tao Li, Tao Qin, Wei Chen, and Tie-Yan Liu. 2020. Learning causal semantic representation for out-of-distribution prediction. arXiv:2011.01681. Retrieved from https://arxiv.org/abs/2011.01681

[58] Yaxu Liu, Jui-Nan Yen, Bowen Yuan, Rundong Shi, Peng Yan, and Chih-Jen Lin. 2022. Practical counterfactual policy learning for Top-K. recommendations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1141–1151.

[59] Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. 2017. Causal effect inference with deep latent-variable models. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 6449–6459.

[60] Hongyu Lu, Min Zhang, and Shaoping Ma. 2018. Between clicks and satisfaction: Study on multi-phase user preferences and satisfaction for online news reading. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 435–444.

[61] G. M. Lunardi, G. M. Machado, V. Maran, and JPMD Oliveira. 2020. A metric for Filter Bubble measurement in recommender algorithms considering the news domain. *Applied Soft Computing* 97, Part A (2020).

[62] James McInerney, Brian Brost, Praveen Chandar, Rishabh Mehrotra, and Benjamin Carterette. 2020. Counterfactual evaluation of slate recommendations with sequential reward interactions. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1779–1788.

[63] Raha Moraffah, Mansooreh Karami, Ruocheng Guo, Adrienne Raglin, and Huan Liu. 2020. Causal interpretability for machine learning-problems, methods and evaluation. *ACM SIGKDD Explorations Newsletter* 22, 1 (2020), 18–33.

[64] Shanlei Mu, Yaliang Li, Wayne Xin Zhao, Jingyuan Wang, Bolin Ding, and Ji-Rong Wen. 2022. Alleviating spurious correlations in knowledge-aware recommendations through counterfactual generator. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[65] Michael P. O'Mahony, Neil J. Hurley, and Guénolé C.M. Silvestre. 2006. Detecting noise in recommender system databases. In *Proceedings of the 11th International Conference on Intelligent User Interfaces*. Association for Computing Machinery, New York, NY, 109–115.

[66] E. Pariser. 2011. *The Filter Bubble: What the Internet is Hiding from You*. penguin UK.

[67] E. Pariser. 2011. *The Filter Bubble: What the Internet is Hiding from You.* The Filter Bubble: What the Internet Is Hiding from You.
[68] J. Passe, C. Drake, and L. Mayger. 2017. Homophily, echo chambers, & selective exposure in social networks: What should civic educators do? *Journal of Social Studies Research* (2017).
[69] Judea Pearl. 1995. Causal diagrams for empirical research. *Biometrika* 82, 4 (1995), 669–688.
[70] Judea Pearl. 2009. Causal inference in statistics: An overview. *Statistics Surveys* 3 (2009), 96–146.
[71] Judea Pearl. 2009. *Causality.* Cambridge University Press.
[72] Judea Pearl and Dana Mackenzie. 2018. *The Book of Why: The New Science of Cause and Effect* (1st. ed.). Basic Books, Inc.
[73] Joseph Ramsey, Madelyn Glymour, Ruben Sanchez-Romero, and Clark Glymour. 2017. A million variables and more: The fast greedy equivalence search algorithm for learning high-dimensional graphical causal models, with an application to functional magnetic resonance images. *International Journal of Data Science and Analytics* 3, 2 (2017), 121–129.
[74] Steffen Rendle. 2010. Factorization machines. In *Proceedings of the 2010 IEEE International Conference on Data Mining*. IEEE, 995–1000.
[75] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the UAI*. 452–461.
[76] Donald B. Rubin. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology* 66, 5 (1974), 688.
[77] Noveen Sachdeva, Yi Su, and Thorsten Joachims. 2020. Off-policy bandits with deficient support. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 965–975.
[78] Yuta Saito. 2020. Asymmetric tri-training for debiasing missing-not-at-random explicit feedback. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 309–318.
[79] Yuta Saito. 2020. Doubly robust estimator for ranking metrics with post-click conversions. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 92–100.
[80] Yuta Saito and Thorsten Joachims. 2021. Counterfactual learning and evaluation for recommender systems: Foundations, implementations, and recent advances. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 828–830.
[81] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.
[82] Masahiro Sato. 2021. Online evaluation methods for the causal effect of recommendations. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 96–101.
[83] Masahiro Sato, Janmajay Singh, Sho Takemori, Takashi Sonoda, Qian Zhang, and Tomoko Ohkuma. 2019. Uplift-based evaluation and optimization of recommenders. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 296–304.
[84] Masahiro Sato, Sho Takemori, Janmajay Singh, and Tomoko Ohkuma. 2020. Unbiased learning for the causal effect of recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems*. 378–387.
[85] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *Proceedings of the International Conference on Machine Learning*. PMLR, 1670–1679.
[86] Tobias Schnabel, Adith Swaminathan, A. Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. arXiv:1602.05352. Retrieved from https://arxiv.org/abs/1602.05352
[87] Gideon Schwarz. 1978. Estimating the dimension of a model. *The Annals of Statistics* (1978), 461–464.
[88] Uri Shalit, Fredrik D. Johansson, and David Sontag. 2017. Estimating individual treatment effect: Generalization bounds and algorithms. In *Proceedings of the International Conference on Machine Learning*. PMLR, 3076–3085.
[89] Jing-Cheng Shi, Yang Yu, Qing Da, Shi-Yong Chen, and An-Xiang Zeng. 2019. Virtual-taobao: Virtualizing real-world online retail environment for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 4902–4909.
[90] Zihua Si, Xueran Han, Xiao Zhang, Jun Xu, Yue Yin, Yang Song, and Ji-Rong Wen. 2022. A model-agnostic causal learning framework for recommendation using search data. In *Proceedings of the ACM Web Conference 2022*. 224–233.
[91] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. 2019. Autoint: Automatic feature interaction learning via self-attentive neural networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 1161–1170.
[92] Peter Spirtes. 2001. An anytime algorithm for causal inference. In *Proceedings of the International Workshop on Artificial Intelligence and Statistics*. PMLR, 278–285.

[93] Peter Spirtes, Clark N. Glymour, Richard Scheines, and David Heckerman. 2000. *Causation, Prediction, and Search*. MIT press.
[94] Harald Steck. 2013. Evaluation of recommendations: Rating-prediction and ranking. In *Proceedings of the 7th ACM Conference on Recommender Systems*. 213–220.
[95] Adith Swaminathan, Akshay Krishnamurthy, Alekh Agarwal, Miro Dudik, John Langford, Damien Jose, and Imed Zitouni. 2017. Off-policy evaluation for slate recommendation. *Advances in Neural Information Processing Systems* 30 (2017).
[96] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. *Proceedings of the 30th ACM International Conference on Information and Knowledge Management* (2021).
[97] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. 1784–1793.
[98] Philip Thomas and Emma Brunskill. 2016. Data-efficient off-policy policy evaluation for reinforcement learning. In *Proceedings of the International Conference on Machine Learning*. PMLR, 2139–2148.
[99] Ha Xuan Tran, Thuc Duy Le, Jiuyong Li, Lin Liu, Jixue Liu, Yanchang Zhao, and Tony Waters. 2021. Recommending the most effective intervention to improve employment for job seekers with disability. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3616–3626.
[100] Nan Wang, Hongning Wang, Yiling Jia, and Yue Yin. 2018. Explainable recommendation via multi-task learning in opinionated text data. *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*.
[101] Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. 2021. Denoising implicit feedback for recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. 373–381.
[102] Wenjie Wang, Fuli Feng, Xiangnan He, Xiang Wang, and Tat-Seng Chua. 2021. Deconfounded recommendation for alleviating bias amplification. In *Proceedings of the KDD'21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. 1717–1725.
[103] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In *Proceedings of the SIGIR'21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11–15, 2021*. 1288–1297.
[104] Wenjie Wang, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2022. User-controllable recommendation against filter bubbles. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.
[105] Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, Min Lin, and Tat-Seng Chua. 2022. Causal representation learning for out-of-distribution recommendation. In *Proceedings of the ACM Web Conference 2022*.
[106] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.
[107] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In *Proceedings of the International Conference on Machine Learning*. PMLR, 6638–6647.
[108] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2021. 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*. 427–435.
[109] Zhenlei Wang, Shiqi Shen, Zhipeng Wang, Bo Chen, Xu Chen, and Ji-Rong Wen. 2022. Unbiased sequential recommendation with latent confounders. In *Proceedings of the ACM Web Conference 2022*. 2195–2204.
[110] Zhenlei Wang, Jingsen Zhang, Hongteng Xu, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Counterfactual data-augmented sequential recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 347–356.
[111] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1791–1800.
[112] Hongyi Wen, Longqi Yang, and Deborah Estrin. 2019. Leveraging post-click feedback for content recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 278–286.
[113] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. 2022. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022), 1–1.

[114] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019. A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 235–244.

[115] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019. A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 235–244.

[116] Peng Wu, Haoxuan Li, Yuhao Deng, Wenjie Hu, Quanyu Dai, Zhenhua Dong, Jie Sun, Rui Zhang, and Xiao-Hua Zhou. 2022. On the opportunity of causal learning in recommendation systems: Foundation, estimation, prediction and challenges. IJCAI.

[117] Lianghao Xia, Yong Xu, Chao Huang, Peng Dai, and Liefeng Bo. 2021. Graph meta network for multi-behavior recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 757–766.

[118] Yikun Xian, Zuohui Fu, Shan Muthukrishnan, Gerard De Melo, and Yongfeng Zhang. 2019. Reinforcement knowledge graph reasoning for explainable recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 285–294.

[119] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. 2017. Attentional factorization machines: learning the weight of feature interactions via attention networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 3119–3125.

[120] Teng Xiao and Suhang Wang. 2022. Towards unbiased and robust causal ranking for recommender systems. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*. 1158–1167.

[121] Xu Xie, Zhaoyang Liu, Shiwen Wu, Fei Sun, Cihang Liu, Jiawei Chen, Jinyang Gao, Bin Cui, and Bolin Ding. 2021. CausCF: Causal collaborative filtering for recommendation effect estimation. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. 4253–4263.

[122] Kun Xiong, Wenwen Ye, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, Binbin Hu, Zhiqiang Zhang, and Jun Zhou. 2021. Counterfactual review-based recommendation. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. 2231–2240.

[123] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. 2020. Adversarial counterfactual learning and evaluation for recommender system. arXiv:2012.02295. Retrieved from https://arxiv.org/abs/2012.02295

[124] Shuyuan Xu, Juntao Tan, Zuohui Fu, Jianchao Ji, Shelby Heinecke, and Yongfeng Zhang. 2022. Dynamic causal collaborative filtering. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management*. 2301–2310.

[125] Shuyuan Xu, Juntao Tan, Shelby Heinecke, Jia Li, and Yongfeng Zhang. 2021. Deconfounded causal collaborative filtering. arXiv:2110.07122. Retrieved from https://arxiv.org/abs/2110.07122

[126] Akihiro Yabe, Daisuke Hatano, Hanna Sumita, Shinji Ito, Naonori Kakimura, Takuro Fukunaga, and Ken-ichi Kawarabayashi. 2018. Causal bandits with propagating inference. In *Proceedings of the International Conference on Machine Learning*. PMLR, 5512–5520.

[127] Mengyue Yang, Quanyu Dai, Zhenhua Dong, Xu Chen, Xiuqiang He, and Jun Wang. 2021. Top-N recommendation with counterfactual user preference simulation. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. 2342–2351.

[128] Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. 2020. A survey on causal inference. arXiv:2002.02770. Retrieved from https://arxiv.org/abs/2002.02770

[129] Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. 2021. A survey on causal inference. *ACM Transactions on Knowledge Discovery from Data* 15, 5 (2021), 1–46.

[130] Shota Yasui, Gota Morishita, Fujita Komei, and Masashi Shibata. 2020. A feedback shift correction in predicting conversion rates under delayed feedback. In *Proceedings of the Web Conference 2020*. 2740–2746.

[131] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. arXiv:1806.01973. Retrieved from https://arxiv.org/abs/1806.01973

[132] Mengqi Zhang, Shu Wu, Xueli Yu, Qiang Liu, and Liang Wang. 2022. Dynamic graph neural networks for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[133] Shengyu Zhang, Dong Yao, Zhou Zhao, Tat-Seng Chua, and Fei Wu. 2021. Causerec: Counterfactual user sequence synthesis for sequential recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 367–377.

[134] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys* 52, 1 (2019), 1–38.

[135] Wenhao Zhang, Wentian Bao, Xiao-Yang Liu, Keping Yang, Quan Lin, Hong Wen, and Ramin Ramezani. 2020. Large-scale causal approaches to debiasing post-click conversion rate estimation with multi-task learning. In *Proceedings of the Web Conference 2020*. 2775–2781.

[136] Weifeng Zhang, Jingwen Mao, Yi Cao, and Congfu Xu. 2020. Multiplex graph neural networks for multi-behavior recommendation. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. 2313–2316.

[137] Xiao Zhang, Haonan Jia, Hanjing Su, Wenhan Wang, Jun Xu, and Ji-Rong Wen. 2021. Counterfactual reward modification for streaming recommendation with delayed feedback. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 41–50.

[138] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal intervention for leveraging popularity bias in recommendation. In *Proceedings of the SIGIR'21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*. 11–20.

[139] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[140] Yu Zheng, Chen Gao, Jianxin Chang, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2022. Disentangling long and short-term interests for recommendation. In *Proceedings of the ACM Web Conference 2022*. 2256–2267.

[141] Yu Zheng, Chen Gao, Xiang Li, Xiangnan He, Yong Li, and Depeng Jin. 2021. Disentangling user interest and conformity for recommendation with causal embedding. In *Proceedings of the Web Conference 2021*. ACM, 2980–2991.

[142] Shengyu Zhu, Ignavier Ng, and Zhitang Chen. 2019. Causal discovery with reinforcement learning. arXiv:1906.04477. Retrieved from https://arxiv.org/abs/1906.04477

[143] Tianyu Zhu, Leilei Sun, and Guoqing Chen. 2021. Graph-based embedding smoothing for sequential recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[144] Xinyuan Zhu, Yang Zhang, Fuli Feng, Xun Yang, Dingxian Wang, and Xiangnan He. 2022. Mitigating hidden confounding effects for causal recommendation. arXiv:2205.07499. Retrieved from https://arxiv.org/abs/2205.07499

[145] Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. arXiv:1906.04571. Retrieved from https://arxiv.org/abs/1906.04571

Received 25 August 2022; revised 27 September 2023; accepted 8 November 2023