Causal Inference in Recommender Systems: A Survey and Future Directions

CHEN GAO, Department of Electronic Engineering, 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, School of Information Science and Technology, University of Science and Technology of China, China
YONG LI, Department of Electronic Engineering, Tsinghua University, China

Recommender systems play a crucial role in information filtering nowadays. Existing recommender systems extract the user preference based on learning the correlation in data, such as behavioral correlation in collaborative filtering, feature-feature, or feature-behavior correlation in click-through rate prediction. However, regretfully, the real world is driven by *causality* rather than correlation, and correlation does not imply causation. For example, the recommender systems can recommend a battery charger to a user after buying a phone, in which the latter can serve as the cause of the former, and such a causal relation cannot be reversed. Recently, to address it, researchers in recommender systems have begun to utilize causal inference to extract causality, enhancing the recommender system. In this survey, we comprehensively review the literature on causal inference-based recommendation. At first, we present the fundamental concepts of both recommendation and causal inference as the basis of later content. We raise the typical issues that the non-causality recommendation is faced. Afterward, we comprehensively review the existing work of causal inference-based recommendation, based on a taxonomy of what kind of problem causal inference addresses. Last, we discuss the open problems in this important research area, along with interesting future works.

Additional Key Words and Phrases: Recommender Systems; Causal Inference; Information Retrieval

1 INTRODUCTION

In the era of information overloading, recommender systems (RecSys) have become the fundamental service for facilitating users' information access. From the early shallow models [40, 63], to recent advances of deep learning-based ones [13, 27], to the most recent graph neural network based models [25, 114], 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 context information. Here, the interaction is mainly induced by the previous recommender system and largely affected by the recommendation policy. Then recommender systems filter from the item-candidates 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, shorten as CTR prediction). Collaborative filtering focuses on users' historical behaviors, such as clicking,

Authors' addresses: Chen Gao, Department of Electronic Engineering, Tsinghua University, China; Yu Zheng, Department of Electronic Engineering, Tsinghua University, China; Wenjie Wang, School of Computing, National University of Singapore, Singapore, chgao96@gmail.com,y-zheng19@mails.tsinghua.edu.cn,wenjiewang96@gmail.com,fulifeng93@gmail.com,xiangnanhe@gmail.com,liyong07@tsinghua.edu.cn; Fuli Feng, School of Information Science and Technology, University of Science and Technology of China, China; Xiangnan He, School of Information Science and Technology, University of Science and Technology of China, China; Yong Li, Department of Electronic Engineering, Tsinghua University, China.

purchasing, etc. 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 uses inner-product to calculate the relevance scores between users and items. To improve the model capacity, recent work [13, 27] takes advantage of deep neural networks for matching users with items, such as neural collaborative filtering [27] 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 [10, 115], user social network in social recommendation [14, 99], multi-type behaviors in multi-behavior recommendation [18, 101], etc. 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 [20], attention networks in AFM [103], stacked self-attention layers in AutoInt [77], etc.

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 [111]. Causal inference is defined as the process of determining and further leveraging the causal relation based on experimental data or observational data [111]. Two popular and widely-used causal-inference frameworks are the potential outcome framework (Rubin Causal Model) [64], and the structural causal model (SCM) [57, 59]. Rubin's Framework aims to calculate the effect of certain treatments. The structural causal model constructs a causal graph and corresponding structural equations, of which there are a set of variables and structural equations describing the causal relations between variables.

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, etc. [45] 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.

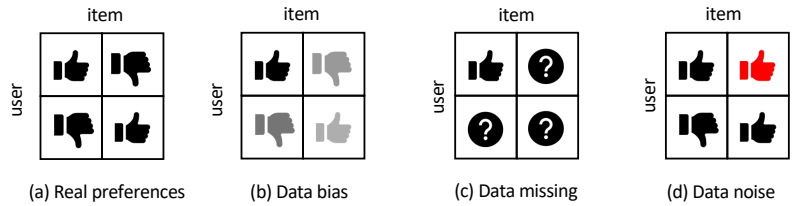

Fig. 1. A simple comparison among three kinds of data issues: data bias, data missing, and data noise, taking collaborative filtering as an example. In general, data bias refers to the biased data collection (*e.g.*, in conformity bias, user behavior does not full relect preferences as it may be due 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.

- **The beyond-accuracy objectives are hard to achieve.** Besides accuracy, recommender systems should also consider other objectives, such as fairness, explainability, transparency, etc. 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 example is diversity. A high-diversity item recommendation list may not be able to fit user interests compared with a high-homogeneity list, for which causality can help capture *why users consume specific category* of items, achieving both accuracy and diversity.

Recent research on recommender systems tackles these challenges with carefully-designed causality-driven methods. There has been a burst of relevant papers in the last two years, and there is a very high probability that causal inference will sweep the field of recommender systems. In this survey paper, we systematically review these early research efforts, especially on how they address the critical shortcomings with causal inference.

First, the recommendation methods with causality can construct the causal graph, under which the bias can be considered as the confounder in most cases, which is further addressed by causal-inference techniques. Second, as for the data missing, the causality-enhanced models can help construct a counterfactual world, and thus the missing can be *collected* via counterfactual reasoning. Third, causal inference can naturally help build interpretable and controllable models, based on which the explainability of both the model itself and the recommendation results can be achieved. Further, other objectives, including diversity, fairness, etc., can also be achieved since the model becomes controllable. Specifically, the current works of causal inference in recommendation can be categorized as follows.

- **Data debiasing with causal inference.** For popularity bias or exposure bias, the bias (due to popularity-aware or exposure strategy-aware data collection) can be regarded as a kind of confounder in most cases. Some existing work addresses it by backdoor adjustment. For conformity bias, it can be described as a collider effect.
- **Data augmentation and data denoising with causal inference.** The two-fold data missing problem includes the limited user-data collection and the recommendation model's causal effect on the system. The extreme scenario of the first fold can even generate the data-noise issue. For the first fold, counterfactual reasoning can help generate the uncollected data as augmentation, which address the data-missing issue. For the second fold, causal models such as IPW can estimate the causal effect of recommendation models.

- **Achieving explainability, diversity, and fairness via interpretable and controllable recommendation model based on causal inference.** Models designed following the causal graph are naturally controllable, of which some representative techniques include causal discovery, disentangled representations, etc. Based on the interpretable model, high diversity can be achieved by controlling the model to avoid the tradeoff, and fair recommendations can be achieved by controlling the model to be fair to specific user demographic.

It is worth mentioning that although there are surveys on either recommender systems [21, 98, 117] or causal inference [22, 51, 51, 112], there is no existing survey discussing this new and important area of causality-driven recommender systems. These surveys of recommender systems mainly introduce and discuss the basic concepts and various advances of recommender systems, with a few discussions on causality-based recommendation. These surveys of causal inference mainly introduce and discuss the basic concepts and fundamental methods of causal inference without enough discussions on applications.

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 question of *why the causal inference is needed and how causal inference enhances recommendation.*
- We provide the necessary knowledge of recommender systems and causal inference, and then dedicated to introducing and explaining the existing work of causal inference for recommendation, from the early attempts and the recently-published papers until 2022.
- 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 system, 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, which involves two representative causal frameworks: SCMs (Structural Causal Models) proposed by Pearl *et al.* [59] and the potential outcome framework developed by Rubin *et al.* [64]. Considering the topic of this survey, we will elaborate on the core concepts by using examples from recommender systems for a clearer understanding. The basic concepts are shown in Fig. 2, which we will explain in detail at the following.

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 [59].

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 Fig. 3, 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 Fig 3 (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 In Fig. 3(b), an item's quality

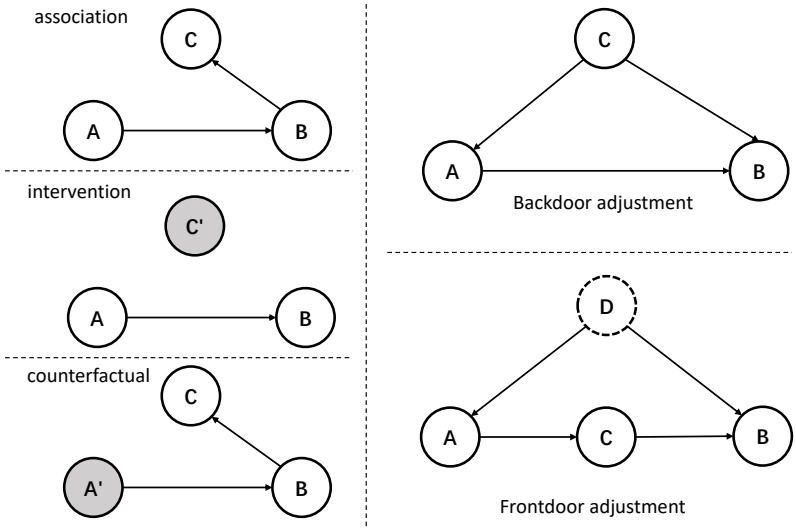

Fig. 2. Important concepts of causal inference.

can affect both its price and users' preferences towards it. In such a fork structure, Z is defined as *confounder variable*. Roughly ignoring confounder Z leads to **incorrent** 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 Fig. 3(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 Z 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-calculas, $do(X = x)$ [59], which blocks the effect of X's parents and set the value of X as x. For example, $do(X = x)$ in Fig. 3(b) will rule out the path $Z \rightarrow X$ and force X to be x [60]. That is, in our above-mentioned example, we set the item prices to a specific value.

Counterfactual. Another important concept is counterfactual, which is the opposite of factual and is used to handle the scenario where the treatment variables' value settings do not happen in the real world. In other words, counterfactual inference estimates what the situation would be if the treatment variable had a different value compared with the observed value in factual world [59]. The key to counterfactual inference is to estimate the values of *error terms* [59] in the factual world and then perform an intervention on the treatment variable to predict the descendant variables in the counterfactual world. For example, a bankrupted seller would be guessing what the sales would be in a counterfactual world where he/she has bought advertisement services by setting treatment variable $T_{\mathbf{if\_ads}} = 1$.

2.1.2 Potential Outcome Framework. The potential outcome framework [64] is another widely-used causal inference framework besides the structural causal model [59]. It estimates the causal effect of a treatment variable on an outcome variable without requiring the causal graph.

Potential Outcome [64]. 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 a potential outcome of Y under one treatment for each individual.

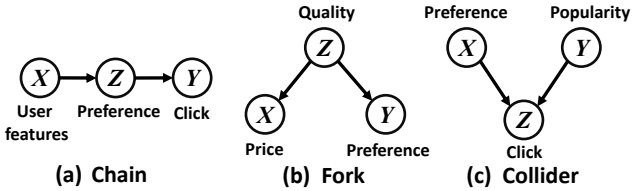

Fig. 3. Illustration of three typical DAGs.

Treatment Effect [64]. 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 $i = \{1, 2, ..., N\}$, ATE is obtained 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 two frameworks. We briefly summarize the similarities and differences between the two frameworks. As stated by Pearl [58], the two frameworks are logically equivalent. The theorem and assumptions in one framework can be equally 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 nice 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 randomly divides users 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 recommendation will hurt user experiences and platform profit. Therefore, it is critical to estimate the causal effect from only observational data. 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 identify causal effects and non-causal effects, such as confounding associations between treatment and outcome. Then the causal effect is distilled by estimation with observed data and the identified causal mechanisms.

One classical method is **backdoor adjustment** [59]. We say a set of variables W satisfies *backdoor criterion* if it blocks all the backdoor paths from the treatment variable T to the outcome variable 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** [59] 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 ($m \in M$),

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

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 follow,

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

which indicates the probability of receiving the treatment given W. Then the causal effect can be estimated by inverse propensity weighting (**IPW**) [29] on the treatment and control group as follow,

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

All of the above causal effect estimations assume that we already have a causal graph. However, in the real world, we may have no prior knowledge about the causal relationships in collected data. It motivates the problem of causal discovery, where we aim to construct a causal graph from existing data of a set of variables. Traditional approaches identify causal relations by conditional independence tests with extra assumptions such as faithfulness [79]. Score-based algorithms [28, 73] are also proposed to relax the strict assumptions for causal discovery, with a score function measuring the quality of the discovered causal graph compared with observed data. Recently, machine learning approaches have been developed to discover causal relations from large-scale data. For example, Zhu *et al.* [125] utilize reinforcement learning method to find an optimal DAG with respect to a scoring function and penalties on acyclicity. There is a survey [22] 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 problems, 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, Twitter, etc. 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, $y_{u',i'} = 0$. The function $f(\cdot, \cdot)$ denotes the recommendation model. Furthermore, with different auxiliary data as input of side-information, there are two primary tasks in recommendation, *i.e.*, collaborative filtering (CF) and click-through rate (CTR) prediction. Desipte the vallila CF which only considers user-item interaction data, some recommendation tasks enhance the behaviral data with auxiliary data, such as social network in social recommendation [15, 100], behavioral sequences in sequential recommendation [7, 126], multiple-type behaviors in multi-behavior recommendation [35, 119], cross-domain user behaviors in cross-domain recommendation [17, 31], etc. For CTR, fine-grained user and item features are important data input,

such as user profiles (occupation, age) and item attributes (category, brand). The mainstream works focus on mapping the high-order cross-features into user-item interactions, for which many models are utilized, such as multi-layer perceptrons [20], attentive neural network [20], self-attentive layers [77], etc.

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 matrix factorization (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** [40]. The similarity function is the inner product as follows,

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

- **NCF** [27]. 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, and $\odot$ indicates the Hadamard product. $\mathbf{h}$ is the weight vector and $\phi(\cdot)$ denotes MLP.
- **NGCF** [91]. This GNN-based recommendation model conducts multiple layers of message passing on the user-item bipartite graph. Formally, the similarity is calculated as follows,

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

  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.

Click-Through Rate Prediction. As introduced above, the unified procedure of CTR prediction is mapping features to user-item interactions. The input features are denoted as follows,

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

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

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

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

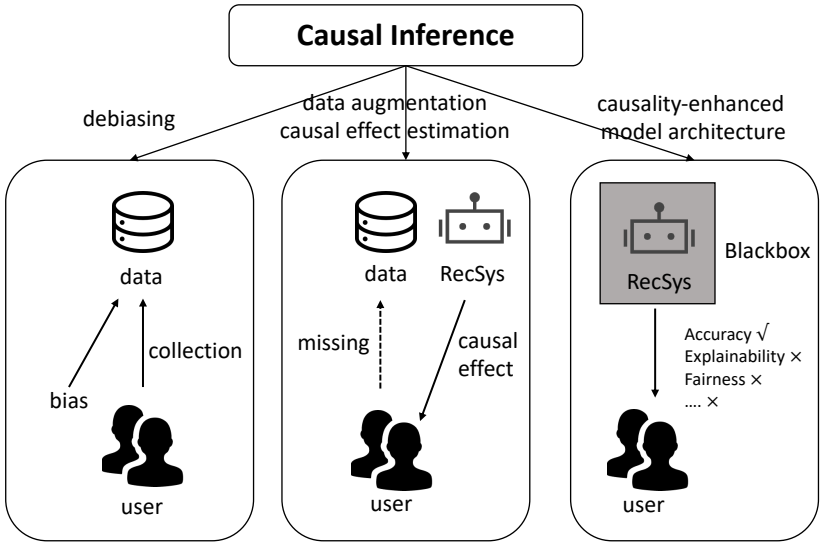

Fig. 4. Illustration of three typical issues of non-causality recommendation models and how causal inference addresses them.

of users' and items' id, 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,

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

The design of $g(\cdot)$ will introduce a module of feature interactions for more powerful correlation learning, via the inner product in FM [62], multi-layer perceptions in DeepFM [20], stacked self-attention layers in AutoInt [77], etc.

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.

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 famous BPR [63] loss is as follows,

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

where $\sigma(\cdot)$ denotes the sigmoid function, and Q is 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 Fig. 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** [96, 121]. 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 [87]. 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 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 [88]. 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 (Fig. 3(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 popular items, leading to over-recommendation, *i.e.,* popularity bias. In the causal dictionary, 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.

[1]https://facebook.com

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 to buy 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 recommender system [65]. Users can interact with specific items only if these items are exposed to them, which strongly correlates with the recommender system's intrinsic strategy. In addition, users may refuse to give feedback [92]. For example, on movie rating websites such as Douban[2], users may only rate a few of the movies they have seen. 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 [71, 80, 83] without causal inference were developed to address the data-missing problem. Steck [80] computes prediction errors for missing ratings. Schnabel *et al.* and Thomas *et al.* [71, 83] 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.* [53] 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.* [43] 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.* [37] points out that users can hardly accurately measure their preferences, thus leading to mismatch between their preferences and final ratings. Cosley *et al.* [12] found that noisy data arises when users map their opinions into discrete ratings. Zhang *er al.* [120] 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 [32, 49, 86, 97] 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-trigger feedback such as click-by-mistake.

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

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.* [12] shows 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 MAE differences arise when exploiting CF models on the original rating data and new rating data. Amatriain *et al.* [2] shows 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.* [86] found through experiments on two representative datasets the performance of recommender system trained by noisy data get 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, the existing solutions are a few. As for causal inference, it can provide a more thorough and explainable user modeling, which can help detect noisy interactions. Specifically, counterfactual reasoning can generate reliable labels by imagining that there is no noise during the data collection.

3.3 Beyond-accuracy Concerns in Recommender Systems

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

3.3.1 Explainability. The requirement of explainability for recommender systems refers to the fact 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 [9, 85, 122] mainly took some item aspects to give explanations, concluded as the aspect-aware explainable recommendation. For example, Wang *et al.* [85] learned users' preferences on given aspects by factorization method to get the aspect-aware explanations.

The necessity of causal inference. Despite effectiveness to some extent, existing methods of explainable recommendation are still limited [81]. Specifically, the explanation is built with 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 component of the recommendation model with the prediction scores. Additionally, the expalnation on recommendation results should fully consider how different decision-factors, *i.e.*, cause, leads 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 that people tend to be isolated from diverse content and information by online personalization [54]. As a consequence, users are placed in a fixed environment in which they can only contact similar topics or information.

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

Passe *et al.* [56] assign 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 [55], which inevitably leads to the homogenous recommendation. Gabriel Machado Lunardi *et al.* [50] empirical analyzed the filter-bubble formation based on popular CF methods and algorithms for diversified recommendation. In terms of human nature, researchers found that people wish to pursue a comfort zone and stay with the opinion that they have interests in or agree [5]. 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 the biased data will exacerbate the homogeneity in recommendation exposure and further the collected data. Moreover, the accuracy-diversity dilemma is another challenge, which refers to the phenomenon that 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 in collected data, supporting exploring 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 recommendation has attracted more attention. As we know, recommender systems serve as a multi-stakeholder platform; thus, the fairness concern of which includes various aspects, including user-side and item-side [6].

The fairness problem on the user side is that different users have different demands for fairness concerns. For example, some users may care more about whether they will be treated unfairly because of their gender, while others may concern about their age on this issue [44]. To enhance users' trust in the recommender system, the user-side fairness problem is required to be addressed in a personalized way. There have been some works [24] trying to solve the fairness issue by association-based methods, whose goal is to eliminate the discrepancy of statistical metrics between different groups. Nevertheless, some research indicates that these association-based methods have some defects and inconsiderate issues [39, 41]. In the process of fairness modeling, association-based methods pay no attention to exploring the link between the objective feature and the outputs of the model. On the contrary, some works reexamine the fairness problem from a causal perspective, which can give us a better understanding of how the outputs change with input variables [1, 38].

As for the item-side fairness issue, it is defined whether each item is equally treated when being recommended. Some possible reasons include the bias or missing of specific items or attributes. To address the item-side fairness problem, some existing work [19, 72] implemented unbiased learning or heuristic ranking adjustment.

The necessity of causal inference. Addressing the fairness problem is similar to answering a question in the counterfactual world if a user does not belong to a certain group or an item does not have a particular attribute, will the recommendation results be the same, or what will the recommendation be? The difference between the counterfactual world and the factual world is the key to fairness evaluation of the recommendation model. Therefore, causal inference-based methods, especially counterfactual methods, can improve the fairness of recommendation from a brand new perspective compared with the existing non-causality methods.

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

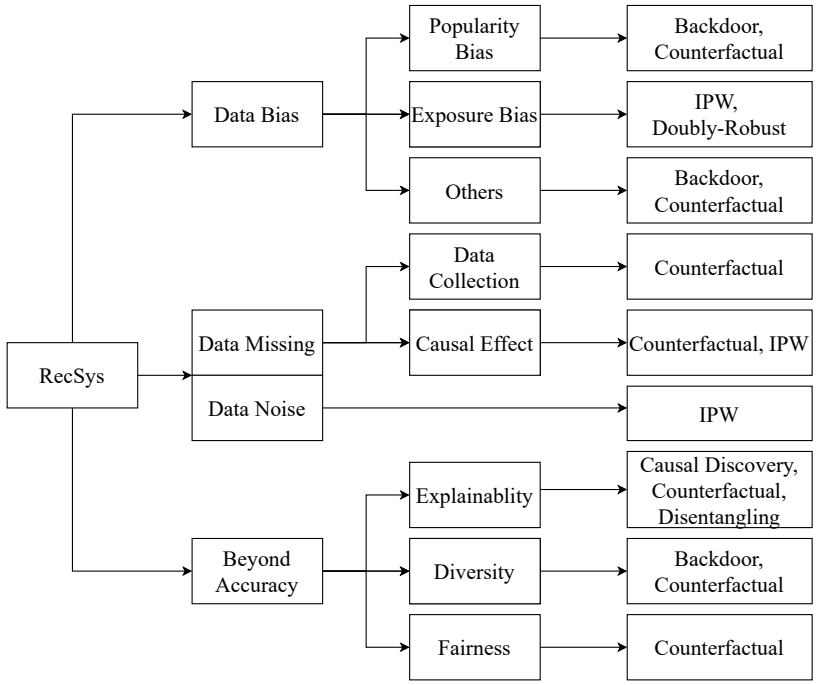

Fig. 5. Illustration of existing work of causal inference for recommendation.

how these challenges can be addressed (at least partly addressed) by presenting the recent advances in the causality-enhanced recommendation.

4 EXISTING WORK OF CAUSAL INFERENCE FOR RECOMMENDATION

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 Fig. 5, and the details are introduced one by one as follows.

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 counterfacual 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.
Strucure 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 [59], which forcibly adjusts the distribution of treatment variables and cuts off the backdoor path from treatment variables to outcome variables via confounders.

Table 1. Representative methods that utilize causal inference to address data bias.

Category	Model	Causal-inference Method	Venue	Year
Popularity Bias	PD [121]	Backdoor Adjustment	SIGIR	2021
	MACR [96]	Counterfactual Inference	KDD	2021
Clickbait Bias	CR [88]	Counterfactual Inference	SIGIR	2021
Bias Amplification	DecRS [87]	Backdoor Adjustment	KDD	2021
Conformity Bias	DICE [124]	Disentangled Causal Embeddings	WWW	2021
Exposure Bias	IPS [72]	IPW	ICML	2016
	Rel-MF [67]	IPW	WSDM	2020
	Multi-IPW/DR [118]	IPW, DR	WWW	2020
	MF-DR-JL [92]	DR	ICML	2019
	DR [66]	DR	RecSys	2020
	MRDR [23]	DR	SIGIR	2021
	LTD [93]	RCT, DR	SIGIR	2021
	AutoDebias [8]	RCT	SIGIR	2021
	USR [94]	IPW	WWW	2022

For example, Zhang *et al.* [121] 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 leverage popularity bias. Besides, Wang *et al.* [87] 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 more action movies to users if they have interacted with a large proportion of action movies before. To tackle this, Wang *et al.* [87] found that the imbalanced item distribution 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 [59]. Xu *et al.* [108] has made some initial attempts to address both global and personalized confounders via frontdoor adjustment. Zhu *et al.* [127] gave a more detailed analysis of the conditions to apply the frontdoor adjustment in recommendation.

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 denote the exposure operation where $o_{u,i} = 1$ means item i is recommended to user u. According to the definition of IPW [46], we can learn a recommender to estimate the causal effect of X on Y by minimizing the following objective,

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

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.* [72] 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 [67], Zhang *et al.* integrated the learning of propensity model and recommendation model into a multi-task learning framework [118], which demonstrates advantages over the separately learned one. Wang *et al.* [94] took the pioneering step of considering the exposure bias in the

sequential recommendation, by proposing an IPW-based method 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 [23, 66, 92] 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 [92] 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 [8, 93] investigates data aggregation strategies, which largely focuses on tackling the sparsity issue of experimental data since RCT is costly.

4.1.2 Colliding Effect. We can discover many collider structures (*cf.* Fig 3(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 [124] 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, which eliminates the path-specific causal effect for debiasing recommendation. In particular, some user/item features might cause shortcuts for interaction prediction, hindering the accurate preference estimation. The 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.* [88] conducted counterfactual inference to remove the effect of exposure features (*e.g.,* attractive titles) for mitigating clickbait issues. In addition, Wei *et al.* [96] reduced the direct causal effect from the item node to the ranking score to alleviate popularity bias. Furthermore, Xu *et al.* [107] proposed an adversarial component to capture the counterfactual exposure mechanism and optimized the candidate model over the worst case 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

Table 2. Representative methods that utilize causal inference to address data missing and data noise.

Category	Model	RecSys Task	Causal-inference Method	Venue	Year
Data Missing	CauseRec [116]	Sequential	Counterfactual	SIGIR	2021
	CASR [95]	Sequential	Counterfactual	SIGIR	2021
	CF^2 [106]	Feature-based	Counterfactual	CIKM	2021
	ASCKG-CG [52]	KG-based	Counterfactual	SIGIR	2022
	CPR [110]	Collaborative Filtering	SCM, Counterfactual	CIKM	2021
	ULO [69]	Collaborative Filtering	Uplift, IPW	RecSys	2019
	DLCE [70]	Collaborative Filtering	IPW	RecSys	2020
	CBI [68]	Collaborative Filtering	Interleaving, IPW	RecSys	2021
	CausCF [105]	Collaborative Filtering	Uplift, RDD	CIKM	2021
	DRIB [104]	Collaborative Filtering	Doubly-Robust, IPW	WSDM	2022
	COR [90]	CTR	Counterfactual	WWW	2022
Data Noise	CBDF [120]	Streaming	Importance Sampling	SIGIR	2021

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 several research fields like computer vision [16], and natural language processing [128]. 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?" [110].
- **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?"[95, 116].
- **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 be more fine-grained from item level to feature level. 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?" [106]. Wang *et al.* [90] 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

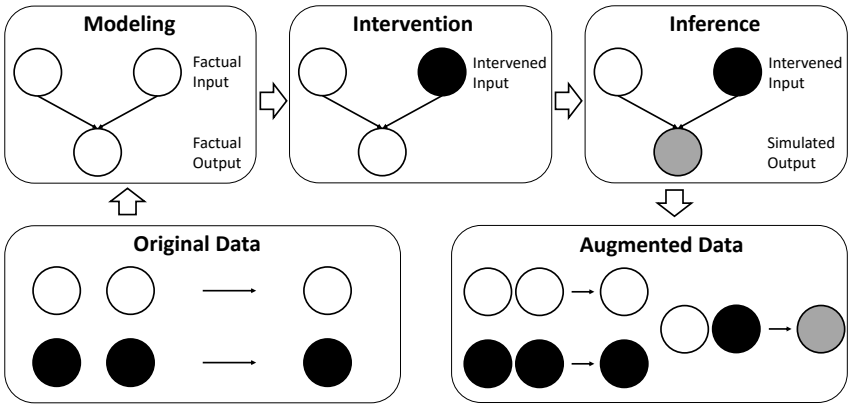

Fig. 6. Illustration of counterfactual data augmentation for data missing.

the counterfactual distribution. A recent work [52] 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.

For all the above three scenarios, counterfactual data augmentation follows a similar paradigm of three steps, which are modeling, intervention, and inference. Fig 6 provides a brief illustration of counterfactual data augmentation, and we 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.* [110] 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.* [106] 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.* [116] and Wang *et al.* [95] 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.

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 [95], a counterfactual interaction sequence can be generated by replacing an item at a random index with a random item. In [116], 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 [110], a counterfactual recommendation list is generated by selecting items with larger loss value *i.e.* the hard samples. In [95] and [106], 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 [110], the counterfactual *clicked* items of the intervened recommendation list are generated by inferring according to the constructed SCM. In [116] and [95], the intervened interaction sequences are fed into the sequential backbone model, and the obtained outputs can directly serve as the counterfactual user embeddings [116], or they can be used to derive counterfactual next items [95].

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, such delayed feedback can be challenging since 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.* [120] tackle the above problem of delayed feedback with the help of causal inference. Specifically, they utilize importance sampling [4, 113] 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.* contributed a colliding inference strategy [86], which leverages the colliding effect [59] of reliable feedback on the predictions to facilitate the users with sparse reliable feedback.

4.2.3 Causal Effect Estimation for Recommendation. As mentioned above, the recommender systems impact the data collection, resulting in missing the real interaction data. Existing recommendation approaches are mainly evaluated and trained over interaction data, and usually, more interactions on the recommended items indicate a more successful recommendation. However, they ignore the fact that some items may have been interacted with by the users even without recommendation. Take e-commerce recommendation as an example, users can have clear intentions and directly purchase the items they want. On the contrary, some items are more effective with respect to recommendation, which means that users will purchase these items if recommended, but will not purchase them if not recommended. Therefore, the purchase probability of these effective items is promoted by recommender systems, *i.e.* uplift. These items reflect a stronger causal effect of recommendation, and it is critical to recommend more items with larger uplift.

A few studies [68–70, 105] in recent years have investigated the causal effect of recommendation from the perspective of uplift. Sato *et al.* [69] apply the potential outcome framework to obtain the average treatment effect (ATE) of recommendation. All the interactions are divided into four categories according to the treatment (recommendation) and the effect (feedback), and then a sampling approach ULO is proposed to learn the uplift of each sample. IPW is adopted to achieve unbiased offline learning [70] and online evaluation [68] on the causal effect of recommendation. Xie *et al.* [105] 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 *el al.* [104] proposed a doubly-robust estimator, along with which a a deep variational information bottleneck method is proposed to aid the adjustment of causal effect estimation.

Table 3. Representative methods that utilize causal inference to achieve beyond-accuracy objectives.

Category	Model	RecSys Task	Causal-inference Method	Venue	Year
Explanability	PGPR [102]	KG-enhanced	Causal Discovery	SIGIR	2019
	CountER [82]	CF	Counterfactual & Causal Discovery	CIKM	2021
	MCT [84]	CTR	Couterfactual	KDD	2021
	CLSR [123]	Sequential	Disentangled Embedding	WWW	2022
	IV4Rec [76]	CTR	Decomposed Embeddings	WWW	2022
Diversity	DecRS [87]	CF	Backdoor Adjustment	KDD	2021
	UCRS [89]	CTR	Counterfactual	SIGIR	2022
Fairness	CBDF [120]	CTR	Counterfactual	SIGIR	2021

4.3 Beyond-accuracy RecSys with Causal Inference

As mentioned in Section 3.3, the non-causality recommender systems may fall into the trap of only achieving higher accuracy while ignoring other important objectives, including explainability, fairness, diversity, etc. 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 recommendation rather than only the correlations. We divide the existing work into three categories as follows.

- *Counterfactual learning*. Tan *et al.* [82] proposed CountER for explainable recommendation based on counterfactual reasoning, which explains the recommendation with the difference between factual and counterfactual worlds. Specifically, they proposed an optimization task that aims to find an item with minimal distance to the original item to reverse the recommendation result in the counterfactual world. CountER [82] also used causal discovery techniques to extract causal relations from historical interactions and the recommended item to enhance the explanation.
- *Causal graph-guided representation learning*. Zheng *et al.* [123] proposed to build 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.* [76] proposed to improve the recommendation model's explainability by decomposing model parameters into two parts: causal part and non-causal. It built a model-agnostic framework by using users' search behaviors as an instrumental variable.
- *Causal discovery.* Xian *et al.* [102] proposed to make use of a knowledge graph for an explainable recommendation. Actually, the paths in the knowledge graph are widely-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 the relation *has_brand* and the node *Apple Brand*. Based on the knowledge graph and users' interaction history, the authors proposed to extract causal relations by reinforcement learning. Specifically, the policy function of reinforcement learning is optimized to explicitly select items via paths of knowledge graph, ensuring accuracy and explanation.
  Tran *et al.* [84] approached the problem of explanable 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 a counterfactual reasoning method that finds the most important feature, of which the modification can lead to employment, was proposed. Therefore, the explanations of employment are the found feature.

4.3.2 Causal Inference for Improving Diversity and Alleviating Filter Bubble. As mentioned above, only pursuing accuracy suffers from the problem of too-homogeneous content, which leads to the so-called filter bubble. Based on causal inference, which helps betters understand and explicitly model the causal effect or user-decision factor, recommendation with better diversity or alleviating filter bubble can be achieved.

- *Counterfactual learning.* Wang *et al.* [89] 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 granularities 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.* [87] approached the problem of homogenous 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 collaborative filtering and click-through rate prediction.

4.3.3 Causal Inference for Fairness in Recommendation. The definition of achieving fairness naturally matches the counterfactual world in causal inference. For example, to judge whether a recommender system is fair or not under a certain user profile, a counterfactual-manner question should be: *would the recommendation results change if the certain user profile is changed?* Li *et al.* [44] proposed the concept of counterfactual fairness of recommendation that with modifying the value of a given feature, the distribution of recommendatin probability keeps the same. The authors approach the problem by proposing users' personalized demand for fairness, and the core idea is to obtain feature-independent user embeddings. To achieve it, a filter module is proposed after the embedding layer to remove those information relevant to sensitive features and then obtain the filtered embeddings. The authors then proposed a prediction module which uses filtered embeddings to predict sensitive features, under an adversarial-learning manner along with the main 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 introduced plenty of work introducing causal inference into the recommender systems. However, existing work with pre-defined causal graphs or structural causal models suffers from two main shortcomings. First, the causal relations may be incorrect. Although the recommendation tailored to the causal relations may improve the recommendation performance, hidden variables may exist that are the real causes. Second, these human-crafted causal graphs are quite simple, of which there are only several involved variables, such as the user conformity, user interest, and user behavior in DICE [124], the exposure feature, user/item/context feature, and

prediction score in CR [95]. Nevertheless, users' decision-making processes may involve many factors in real-world scenarios. For example, whether a user visits a restaurant depends on the restaurant's position, taste, brand, price, etc. Therefore, it is essential to design causal discovery methods for learning causality from real-world data in recommender systems. The traditional methods for causal discovery can be divided into the following categories. Constraint-based (CB) algorithms are one of the most representative ones, such as PC algorithm [79], FCI algorithm [78], etc., which first extract conditional independence between variable pairs and then construct a directed acyclic graph based on it. GES methods [11, 61] further extend CB algorithms with a scoring function to evaluate the rationality of a DAG. However, these existing methods are still suffering from the issue of high search costs or low robustness under large-scale data [22]. Recently, deep learning methods [36, 48, 74] and reinforcement learning methods [125] have been further proposed for inferring causal relations from large-scale data. Therefore, it is a promising and crucial future direction for discovering causal relations and then leveraging the learned causality to enhance recommendation.

5.2 Causality-aware Stable and Robust Recommendation

Recommender systems are required 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, etc. As a result, the data distribution may be fast-changing [90]. Second, there are multiple recommendation scenarios, such as different tabs in the same mobile App, multiple domains, or multiple objectives, which means the recommendation model should be robust and stable. Last, there is a gap between offline evaluations and online experiments, among which a recommendation model with good performance in offline experiments should have good online results. As for higher stability and robustness of machine learning models, existing work [34, 47] has demonstrated the causality-aware model is a promising solution, with solid ability in domain adaption or out-of-distribution (OOD) generalization [90]. Therefore, leveraging causality for robust and stable recommendations is very important.

5.3 Causality-aware Evaluation for Recommendation

In Section 4.3, we have discussed how causal inference enhances recommender systems for those beyond-accuracy evaluation metrics, including diversity, fairness, and explainability. Nevertheless, there are plenty of less-explored problems in the evaluation of recommendation. First, in real-world applications, user engagement or satisfaction is the most crucial concern, rather than accuracy or other metrics. Thus, it is pretty important to understand how recommendation models generate results that improve user engagement. Thus, causality should be considered in the evaluation of recommendation models. It is even more challenging than explainability. Second, although many existing works are modeling the exposure bias, missing-not-at-random, or causal effect of recommender systems, trying to erase the gap between offline and online evaluation, the current solutions are still limited. As mentioned above, improving model generalization ability and robustness with causality can partly address the distribution shift in offline and online data. However, the key to this problem is the modeling of how recommender systems affect users' decisions (such as the disentangling of user interests) and how the collective feedback further affects the system, based on the view of causality.

5.4 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 [26] against traditional neural

network models [27] 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. Nevertheless, some critical problems are still unresolved, waiting for causality-supported solutions. First, how do GNNs make an accurate and successful recommendation? The explainability, including the model itself and the recommendation results of powerful GNN-based recommendation models, still needs much research. Currently, they work as a black box. Second, although the recent advances in causality-aware recommendation models may take the GNN module as one of the backbones, the GNN module itself and causal inference are separated. Explicitly coupling the message-passing process and causal inference/reasoning for recommendation is still an open and unexplored research field.

5.5 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 long-term and more reasonable objective, the system should not only estimate the current or next-step user interaction but also consider the sequence of interactions to maximize user engagement or platform requirements. Due to the nature of dynamics and user-system interactions, some works [33, 75] proposed simulators of recommender systems. Specifically, these works leverage reinforcement learning methods, such as imitation learning [30], for simulating how users choose items under a specific environment and context. However, these works are data-driven, lacking the support of causality, which may result in an incorrect decision-making process. Recently, causal reinforcement learning (CRL) methods are proposed to leverage causal inference to address the data missing issue in reinforcement learning tasks. Bareinboim *et al.* [3] proposed to leverage causal interventions to help estimate the reward considering the unobserved confounders. There are other further works [42, 109] on causal bandit algorithms, which provide a theoretical bound of performance improvement against non-causal bandits. In short, causally-aware reinforcement learning approaches have promising model generality on insufficient data in the modeling of dynamics, especially for dynamic and sequential user-system interaction. Thus, they will play an essential role in modeling the simulator and the environment of recommender systems.

To summarize, future works of causality-aware recommender systems should first address the limitation of the pre-defined causal graph. Other promising research directions include improving the robustness, such as domain generalization, better evaluation for long-term utility and erasing the offline-online gap, better fusion with graph neural networks, and causality-supported simulators of recommender systems.

6 CONCLUSION

In very recent years, causal inference has become a very important topic in the research field of recommender systems, which can be said without exaggeration, and has reshaped our recognition of recommendation models. This paper takes the first step to provide a survey of existing work by carefully and systematically discussing why causal inference can and how it addresses the weaknesses of non-causality recommendation models. We hope this survey can well motivate researchers in this area and, more importantly, researchers we plan to start research in this new area.

REFERENCES

[1] Junzhe Zhang and Elias Bareinboim . 2018. Fairness in Decision-Making – The Causal Explanation Formula. In *AAAI 2018*.

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

247–258.

[3] 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.

[4] 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).

[5] 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.

[6] Robin Burke. 2017. Multisided fairness for recommendation. *arXiv preprint arXiv:1707.00093* (2017).

[7] 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.

[8] 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.

[9] 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* (2020).

[10] 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.

[11] David Maxwell Chickering. 2002. Optimal structure identification with greedy search. *Journal of machine learning research* 3, Nov (2002), 507–554.

[12] 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.

[13] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *RecSys*. 191–198.

[14] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. 2019. Graph neural networks for social recommendation. In *The world wide web conference*. 417–426.

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

[16] 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 *European Conference on Computer Vision*. Springer, 71–86.

[17] 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 *The world wide web conference*. 491–502.

[18] 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 *2019 IEEE 35th international conference on data engineering (ICDE)*. IEEE, 1554–1557.

[19] 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 Twelfth ACM International Conference on Web Search and Data Mining* (2019).

[20] 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.

[21] 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).

[22] 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 (CSUR)* 53, 4 (2020), 1–37.

[23] 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.

[24] Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of Opportunity in Supervised Learning. In *NIPS*.

[25] 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.

[26] 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* (2020).
[27] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*. 173–182.
[28] 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.
[29] 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.
[30] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. *Advances in neural information processing systems* 29 (2016).
[31] 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.
[32] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *ICDM*. 263–272.
[33] 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 preprint arXiv:1909.04847* (2019).
[34] Dominik Janzing. 2019. Causal Regularization. *Advances in Neural Information Processing Systems* 32 (2019), 12704–12714.
[35] 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.
[36] Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *International conference on machine learning*. PMLR, 3020–3029.
[37] Nicolas Jones, Armelle Brun, and Anne Boyer. 2011. Comparisons instead of ratings: Towards more stable preferences. In *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology*, Vol. 1. IEEE, 451–456.
[38] Junzhe Zhang and Elias Bareinboim. 2018. Equality of Opportunity in Classification: A Causal Approach. In *NeurIPS*.
[39] 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).
[40] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009).
[41] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual Fairness. In *NIPS*.
[42] 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.
[43] 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.
[44] 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.
[45] 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.
[46] Roderick JA Little and Donald B Rubin. 2019. *Statistical analysis with missing data*. Vol. 793. John Wiley & Sons.
[47] 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 preprint arXiv:2011.01681* (2020).
[48] 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.
[49] 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 *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 435–444.
[50] 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).
[51] 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.

[52] 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 *SIGIR*.

[53] 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 (IUI '06)*. Association for Computing Machinery, New York, NY, USA, 109–115.

[54] Eli Pariser. 2011. *The filter bubble: What the Internet is hiding from you.* penguin UK.

[55] E. Pariser. 2011. *The Filter Bubble: What the Internet Is Hiding from You.* The Filter Bubble: What the Internet Is Hiding from You.

[56] 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).

[57] Judea Pearl. 1995. Causal diagrams for empirical research. *Biometrika* 82, 4 (1995), 669–688.

[58] Judea Pearl. 2009. Causal inference in statistics: An overview. *Statistics surveys* 3 (2009), 96–146.

[59] Judea Pearl. 2009. *Causality.* Cambridge university press.

[60] Judea Pearl and Dana Mackenzie. 2018. *The Book of Why: The New Science of Cause and Effect* (1st ed.). Basic Books, Inc.

[61] 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.

[62] Steffen Rendle. 2010. Factorization machines. In *2010 IEEE International conference on data mining*. IEEE, 995–1000.

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

[64] Donald B Rubin. 1974. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology* 66, 5 (1974), 688.

[65] 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.

[66] Yuta Saito. 2020. Doubly robust estimator for ranking metrics with post-click conversions. In *Fourteenth ACM Conference on Recommender Systems*. 92–100.

[67] 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.

[68] Masahiro Sato. 2021. Online Evaluation Methods for the Causal Effect of Recommendations. In *Fifteenth ACM Conference on Recommender Systems*. 96–101.

[69] 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.

[70] Masahiro Sato, Sho Takemori, Janmajay Singh, and Tomoko Ohkuma. 2020. Unbiased learning for the causal effect of recommendation. In *Fourteenth ACM Conference on Recommender Systems*. 378–387.

[71] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*. PMLR, 1670–1679.

[72] Tobias Schnabel, Adith Swaminathan, A. Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as Treatments: Debiasing Learning and Evaluation. *ArXiv* abs/1602.05352 (2016).

[73] Gideon Schwarz. 1978. Estimating the dimension of a model. *The annals of statistics* (1978), 461–464.

[74] Uri Shalit, Fredrik D Johansson, and David Sontag. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International Conference on Machine Learning*. PMLR, 3076–3085.

[75] 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*, Vol. 33. 4902–4909.

[76] 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.

[77] 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.

[78] Peter Spirtes. 2001. An anytime algorithm for causal inference. In *International Workshop on Artificial Intelligence and Statistics*. PMLR, 278–285.

[79] Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. 2000. *Causation, prediction, and search*. MIT press.
[80] Harald Steck. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Proceedings of the 7th ACM conference on Recommender systems*. 213–220.
[81] 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 & Knowledge Management* (2021).
[82] 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 & Knowledge Management*. 1784–1793.
[83] Philip Thomas and Emma Brunskill. 2016. Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*. PMLR, 2139–2148.
[84] 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 & Data Mining*. 3616–3626.
[85] Nan Wang, Hongning Wang, Yiling Jia, and Yue Yin. 2018. Explainable Recommendation via Multi-Task Learning in Opinionated Text Data. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (2018).
[86] 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.
[87] Wenjie Wang, Fuli Feng, Xiangnan He, Xiang Wang, and Tat-Seng Chua. 2021. Deconfounded Recommendation for Alleviating Bias Amplification. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*. 1717–1725.
[88] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks can be Cheating: Counterfactual Recommendation for Mitigating Clickbait Issue. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*. 1288–1297.
[89] Wenjie Wang, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2022. User-controllable Recommendation Against Filter Bubbles. In *SIGIR*.
[90] 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*.
[91] 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.
[92] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In *International Conference on Machine Learning*. PMLR, 6638–6647.
[93] 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.
[94] 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.
[95] 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.
[96] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1791–1800.
[97] 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.
[98] 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.
[99] 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.
[100] 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.

[101] 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.

[102] 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.

[103] 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.

[104] Teng Xiao and Suhang Wang. 2022. Towards Unbiased and Robust Causal Ranking for Recommender Systems. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. 1158–1167.

[105] 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. 4253–4263.

[106] 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 & Knowledge Management*. 2231–2240.

[107] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. 2020. Adversarial counterfactual learning and evaluation for recommender system. *arXiv preprint arXiv:2012.02295* (2020).

[108] Shuyuan Xu, Juntao Tan, Shelby Heinecke, Jia Li, and Yongfeng Zhang. 2021. Deconfounded Causal Collaborative Filtering. *arXiv preprint arXiv:2110.07122* (2021).

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

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

[111] Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. 2020. A survey on causal inference. *arXiv preprint arXiv:2002.02770* (2020).

[112] 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 (TKDD)* 15, 5 (2021), 1–46.

[113] 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.

[114] 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 preprint arXiv:1806.01973* (2018).

[115] 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).

[116] 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.

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

[118] 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.

[119] 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 & Knowledge Management*. 2313–2316.

[120] 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.

[121] 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 *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*. 11–20.

[122] 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 & development in information retrieval* (2014).

[123] 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.

[124] 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 *WWW*. ACM, 2980–2991.

[125] Shengyu Zhu, Ignavier Ng, and Zhitang Chen. 2019. Causal discovery with reinforcement learning. *arXiv preprint arXiv:1906.04477* (2019).

[126] Tianyu Zhu, Leilei Sun, and Guoqing Chen. 2021. Graph-based Embedding Smoothing for Sequential Recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2021).

[127] Xinyuan Zhu, Yang Zhang, Fuli Feng, Xun Yang, Dingxian Wang, and Xiangnan He. 2022. Mitigating Hidden Confounding Effects for Causal Recommendation. *arXiv preprint arXiv:2205.07499* (2022).

[128] Ran Zmigrod, Sabrina J Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. *arXiv preprint arXiv:1906.04571* (2019).