A Model-Agnostic Causal Learning Framework for Recommendation using Search Data

Zihua Si[1,2], Xueran Han[1,2], Xiao Zhang[1,2], Jun Xu[1,2,*], Yue Yin[3], Yang Song[3], Ji-Rong Wen[1,2]
[1] Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China;
[2] Beijing Key Laboratory of Big Data Management and Analysis Methods; [3] Kuaishou Technology Co., Ltd., China
{2018202181, hanxueran, zhangx89, junxu, jrwen}@ruc.edu.cn; {yinyue, yangsong}@kuaishou.com

ABSTRACT

Machine-learning based recommender systems(RSs) has become an effective means to help people automatically discover their interests. Existing models often represent the rich information for recommendation, such as items, users, and contexts, as embedding vectors and leverage them to predict users' feedback. In the view of causal analysis, the associations between these embedding vectors and users' feedback are a mixture of the causal part that describes why an item is preferred by a user, and the non-causal part that merely reflects the statistical dependencies between users and items, for example, the exposure mechanism, public opinions, display position, etc. However, existing RSs mostly ignored the striking differences between the causal parts and non-causal parts when using these embedding vectors. In this paper, we propose a model-agnostic framework named IV4Rec that can effectively decompose the embedding vectors into these two parts, hence enhancing recommendation results. Specifically, we jointly consider users' behaviors in search scenarios and recommendation scenarios. Adopting the concepts in causal analysis, we embed users' search behaviors as *instrumental variables* (IVs), to help decompose original embedding vectors in recommendation, i.e., *treatments*. IV4Rec then combines the two parts through deep neural networks and uses the combined results for recommendation. IV4Rec is model-agnostic and can be applied to a number of existing RSs such as DIN and NRHUB. Experimental results on both public and proprietary industrial datasets demonstrate that IV4Rec consistently enhances RSs and outperforms a framework that jointly considers search and recommendation.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

recommendation; search; causal learning; instrumental variables

ACM Reference Format:
Zihua Si, Xueran Han, Xiao Zhang, Jun Xu, Yue Yin, Yang Song, Ji-Rong Wen. 2022. A Model-Agnostic Causal Learning Framework for Recommendation using Search Data. In *Proceedings of the ACM Web Conference 2022 (WWW '22), April 25–29, 2022, Virtual Event, Lyon, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3485447.3511951

1 INTRODUCTION

Recommendation and search have become two major approaches to help users to obtain information from the Internet. Traditionally, recommendation and search were usually deployed as two separate systems, serving different users with different types of information objectives. In recent years, many online content platforms provide both search and recommendation services in one application. While having heterogeneous user inputs, these two services can be connected through their common sets of users and items. This phenomenon provides us the opportunity to further improve the performance of one service through using the user activities collected from the other. Early studies have been conducted and showed that jointly optimizing the search and recommendation can improve their respective performances [41, 42].

Traditional recommender systems(RSs) utilize the rich information from the user, the item, and the context to make recommendations. Usually, this information is represented as real-valued embedding vectors. The users' preference to an item, therefore, is calculated based on these embeddings, e.g., using dot product between the user and item embeddings. From the viewpoint of causal analysis, the signals characterized by the embeddings can be decomposed into two parts: the causal association part which describes why a user prefers an item under the context; the non-causal association part, on the other hand, is often affected by many factors, such as the exposure mechanism, public opinions, display position, etc. Thus it merely reflects the statistical dependencies between users and items. The striking differences between causal and non-causal associations lead to their different roles in RSs. While the causal association part contains key signals that lead to the outcomes (e.g., clicks), the non-causal association part may still influence the outcomes through a few unobserved confounders.

Since the causal association part and non-causal association part in embedding vectors affect the final recommendation through different mechanisms, to achieve optimal performance, an ideal RS can employ different approaches to handle corresponding signals respectively. However, existing RSs mostly ignored the differences between the two parts through using the embeddings as a whole. In this paper, we propose a model-agnostic framework named IV4Rec that can effectively decompose the embedding vectors into these two parts by jointly considering users' behavior in search scenarios and recommendation scenarios. Specifically, adopting the concepts in causal analysis, we embed users' search behavior as *instrumental*

[*] Corresponding author: Jun Xu .

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WWW '22, April 25–29, 2022, Virtual Event, Lyon, France
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9096-5/22/04…$15.00
https://doi.org/10.1145/3485447.3511951

variables (IVs), to help decompose original embedding vectors in recommendation, i.e., *treatments*.

In causal inference, IVs methods have been widely used to predict the effects of unobserved causes [11]. After identifying the IVs that only affect the treatments and not the confounders, the IV regression can basically split the treatment variable into two parts: one part that has causal correlation and one part that probably does not. Since the search and recommendation services are deployed in one platform and are with shared user groups and candidate items, users' search activities also reflect their preferences in recommendation scenarios. Therefore, it is reasonable to take users' search activities as IVs to decompose the recommendation embeddings into causal association and non-causal association parts.

In our framework, when considering to recommend an item to a user, a set of queries related to this item is collected as IVs for this item. For example, queries that most users search for before clicking on this item. The IVs are represented by embeddings and are used to fit the original embedding of this item through a regression model. In this way, the original item embedding can be successfully decomposed into the causal part (the values fitted by the regression model) and the non-causal part (the residuals). Finally, these two parts are reconstructed into a new vector (new treatment) and fed into the RS. Since users are usually represented by their browsing histories in RSs, the embedding of each item in a user's history can also be decomposed in this way, hence further enhancing the representation of users.

The contributions of this paper are summarized as follows:

(1) We propose a model-agnostic framework, IV4Rec, to improve recommendation using search data. By considering users' search behavior as IVs to help decompose original embeddings in RS, the framework is able to enhance the representation of both users and items in RSs.

(2) We propose an approach to constructing new treatments through original embeddings and IVs. We use a regression model to decompose original embeddings into a causal part and a non-causal part, which are combined using neural networks and can be jointly trained with any suitable RSs.

(3) We conducted extensive experiments on a public dataset and a real-world industrial dataset. Experimental results demonstrated that IV4Rec can consistently enhance different RSs. In particular, using search activities as IVs for recommendation outperformed traditional methods that jointly model search and recommendation but ignore the causal effect.

2 RELATED WORK

Traditionally, search and recommendation are designed as two separate systems and a large number of search models [6] and RSs [24] have been developed. Garcia-Molina et al. [8] also pointed out that search (information retrieval) and recommendation (information filtering) are the two sides of the same coin. They have strong connections and similarities [35]. Recently, there is a trend to jointly model and optimize the search and recommendation and promote their accuracy at the same time [41, 42]. For example, Zamani and Croft [41] assume that search engines and RSs could potentially benefit from each other and designed a joint learning framework; Zamani and Croft [42] extend the work by joint learning search and recommendation models from user-item interactions; Yao et al. [38] design an approach called USER that mines user interests from the integrated sequence and accomplishes these two tasks in a unified way, and applied to the tasks of personalized search and recommendation.

Besides the joint modeling, methods also have been developed to make use of search or recommendation as the external information to improve the performances of recommendation and search [31, 34, 39]. Wu et al. [34] propose a Zero-Shot Heterogeneous Transfer Learning framework that transfers the learned knowledge from the recommendation component to the search component, addressed the cold-start problem in the search system. Wu et al. [31], Yao et al. [39] use the search history log to enhance the recommendation task as external information. In this paper, we also make use of the search data as external information to enhance the recommendation.

In this paper, we use the instrumental variables (IVs) [4, 12, 29], a popular method in causal inference [14], to enhance recommendation with search data. Most IVs works make use of a two-stage least squares (2SLS) procedure [17]. Recently many IV-based causal learning methods extend 2SLS with deep learning methods. Hartford et al. [11] provide a flexible framework to combine deep learning methods and the 2SLS method. Xu et al. [36] provide an alternating training regime for 2SLS and attain good end-to-end performance in high dimensional image data and off-policy reinforcement learning tasks. Yuan et al. [40] utilize mutual information to learn IV representation and confounder representation, which are used as inputs for two-stage regression with neural networks structure. In recommendation, causal learning has been used for tackling problem of the biases (e.g., position bias, popularity bias, selection bias etc.) [2, 21, 22, 27] and fairness [9, 18, 20]. Many researchers focus on causal embedding for recommendation [3, 15, 33, 45]. They are interested in finding the optimal treatment recommendation policy that maximizes the reward concerning the control recommendation policy for each user [3] or learning a fairness or unbiased representation of items and users for recommendation[15, 33, 45]. Other researchers propose a few methods to fit the preference of users with weighted click data, where each click is weighted by the inverse probability (IPW) of exposure [19, 26, 30, 43].

3 PROBLEM FORMULATION

This section formalizes the problem of recommendation with search queries as IVs.

3.1 Background

3.1.1 Recommendation and search in one platform. A number of content platforms provide both search and recommendation, which serve the same set of users with the same set of items. From the viewpoint of recommendation, when a user $u \in \mathcal{U}$ accesses the platform, the system provides a list of items $i \in \mathcal{I}$ with an existing RS. Often, user u interacts with items in certain context denoted as $\mathbf{p}_u$, including the user profile, search history, or situational context, which can be collected by the platform and represented as real-valued vectors (embeddings) $\mathbf{p}_u \in \mathbb{R}^{d_c}$, where d_c is the dimension of embedding for context. Usually, each user u and each item i can also be represented as real-valued vectors (embeddings), denoted

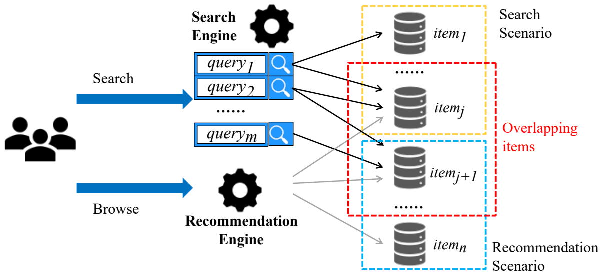


Figure 1: Recommendation and search services in one platform. Search scenario: users issue queries and click returned items. Recommendation scenario: users browse returned items. There exist overlapping items in both services.

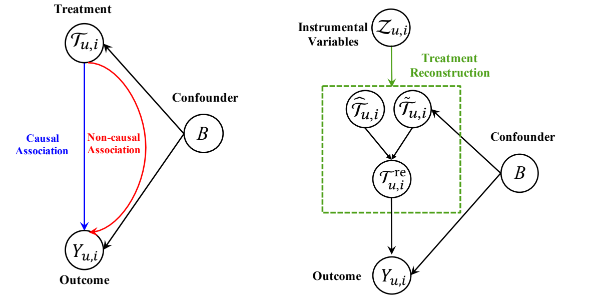
as $\mathbf{t}_u \in \mathbb{R}^{d_u}$ and $\mathbf{t}_i \in \mathbb{R}^{d_i}$, respectively, where d_i and d_u are the dimensions of the embeddings for users and items. The RS is usually trained with the historical user-system interactions $\mathcal{D}^{\text{rec}}$ where each tuple $(u, i, c) \in \mathcal{D}^{\text{rec}}$ means that the item i was shown to the user u and the interaction is $c \in \{0, 1\}$ where $c = 1$ means clicked and $c = 0$ otherwise.

From the viewpoint of search, when a user $u \in \mathcal{U}$ issues a query $q \in \mathcal{Q}$ where q is a text query and $\mathcal{Q}$ is the set of all queries, the system also provides a list of items $i \in \mathcal{I}$ with an existing search model. Similarly, each query can be represented as an embedding vector $\mathbf{t}_q \in \mathbb{R}^{d_q}$, where d_q is the embedding dimension. Since search and recommendation shared the same set of users $\mathcal{U}$ and items $\mathcal{I}$, the user u and item i in search are also represented as the same embeddings $\mathbf{t}_u$ and $\mathbf{t}_i$ which are identical to those in recommendation. The historical user-system interactions in the search can be denoted $\mathcal{D}^{\text{src}}$ where each tuple $(u, q, i, c) \in \mathcal{D}^{\text{src}}$ indicates that a user u is shown with item i after issuing the query q, and the user's activity is $c \in \{0, 1\}$. Since the search and recommendation serve the users with the same set of items, it is inevitable that there exist overlaps between $\mathcal{D}^{\text{rec}}$ and $\mathcal{D}^{\text{src}}$, that is, they have common target items in their records. As shown in Figure 1, there exist overlapping items in search and recommendation scenarios.

3.1.2 Method of instrumental variables. In causal inference, the method of IVs [1, 5] aims to estimate the causal effect between a treatment variable X and an outcome variable Y, in the presence of other variables (e.g., confounders) that are associated with the treatment and outcome simultaneously. Theoretically, a variable Z is a valid *instrumental variable* if it is unconfounded by the confounder (may be unobserved) and only affects the outcome Y via the treatment X. Typical IVs methods such as 2SLS [17] adopt a two-stage least square regression to find the causal effect of treatment X on the outcome Y: first regresses the treatment on the instrument and obtains a reconstructed treatment; then regresses the outcome on the reconstructed treatment from the first stage. An unbiased estimate of causal effect can be achieved from the coefficients of the second stage regression.

3.2 Causal view of recommendation

Existing RSs are usually trained on the user-system historical activities $\mathcal{D}^{\text{rec}}$, with the assumption that the click c in each of the training records $(u, i, c) \in \mathcal{D}^{\text{rec}}$ unbiasedly reflects the preference of u to i. In real world, however, the user clicks recorded in $\mathcal{D}^{\text{rec}}$ can be often affected by many factors (e.g., confounders), including the position bias, selection bias [10], and popularity bias [13], etc. From the viewpoint of causal inference, we can regard the embedding vectors (i.e., the representations of users and items) as the treatment $\mathcal{T}_{u,i}$, and the user's feedback (i.e., click) as the outcome $Y_{u,i}$.



(a) Conventional recommendation models mix the causal and non-causal associations between treatment and outcome.

(b) IV4Rec reconstructs treatment by leveraging IVs to decompose treatment into causal and non-causal parts and combining them with different weights.

Figure 2: (a): conventional RSs. (b): RSs intervened by IVs. $\mathcal{T}_{u,i}$: embeddings of user and item. B: confounders (e.g. popularity bias, selection bias). $Y_{u,i}$: user feedback. $\mathcal{Z}_{u,i}$: IVs (i.e. queries). $\widehat{\mathcal{T}}_{u,i}$: the fitted vectors. $\widetilde{\mathcal{T}}_{u,i}$: the residuals.

Following the framework in [23], a causal graph for conventional RSs can be constructed in Figure 2(a), where conventional RSs simply estimate the mixed associations between the treatment $\mathcal{T}_{u,i}$ and the outcome $Y_{u,i}$. Due to the presence of (unknown) confounders B, there exist two paths from treatment $\mathcal{T}_{u,i}$ to outcome $Y_{u,i}$, including a path of non-causal association that is facilitated by the confounder (the red arrow curve from $\mathcal{T}_{u,i}$ to $Y_{u,i}$), and a path of causal association that describes why an item is preferred by a user (the blue arrow line from $\mathcal{T}_{u,i}$ to $Y_{u,i}$). Specifically, the non-causal association part is affected by confounders, such as the exposure mechanism, public opinions, display position, etc. Thus non-causal and causal associations reflect different relations between user-item pair (i.e., treatments) and user's feedback (i.e., outcome).

It is difficult to identify the causal associations based on the biased observations $\mathcal{D}^{\text{rec}}$ given the unknown number of unknown confounders. Fortunately, the users' search activities in $\mathcal{D}^{\text{src}}$ provide us a chance to decompose the treatment $\mathcal{T}_{u,i}$. As shown in Figure 2(b), we leverage the related queries as IVs, denoted as $\mathcal{Z}_{u,i}$, and regress $\mathcal{T}_{u,i}$ on $\mathcal{Z}_{u,i}$ to get $\widehat{\mathcal{T}}_{u,i}$ which doesn't depend on the confounders B. Thus the relation between $\widehat{\mathcal{T}}_{u,i}$ and $Y_{u,i}$ can be seen as a causal association. We also calculate the residuals $\widetilde{\mathcal{T}}_{u,i}$ of the regression. The relation between $\widetilde{\mathcal{T}}_{u,i}$ and $Y_{u,i}$ can be seen as a non-causal association. Treatments are reconstructed by combining the fitted vectors $\widehat{\mathcal{T}}_{u,i}$ and the residuals $\widetilde{\mathcal{T}}_{u,i}$. Therefore users' search activities are injected into RSs under a causal learning framework.

4 OUR APPROACH: IV4REC

This section describes the proposed IV4Rec framework.

4.1 Model overview

IV4Rec mainly consists of three steps, shown in Figure 3. First, it defines the treatment $\mathcal{T}_{u,i}$ based on the recommendation data $\mathcal{D}^{\text{rec}}$, and constructs the IVs $\mathcal{Z}_{u,i}$ based on the search data $\mathcal{D}^{\text{src}}$. Then, it reconstructs the treatment $\mathcal{T}_{u,i}^{\text{re}}$ through regressing treatment $\mathcal{T}_{u,i}$ on IVs $\mathcal{Z}_{u,i}$. Finally, the reconstructed treatments are fed to a RS.

4.2 Construction of treatments and IVs

To predict the preference score of a target user-item pair (u, i), a treatment variable $\mathcal{T}_{u,i}$ in RS can be defined as a set of embeddings, including the embedding of the target item i and the embeddings of the items interacted with u:

$$\mathcal{T}_{u,i} = \{\mathbf{t}_j : j \in \mathcal{I}_u \cup \{i\}\}, \tag{1}$$

where $\mathbf{t}_j \in \mathbb{R}^{d_i}$ is the embedding vector of item j, which is usually generated by some representation learning methods (e.g., BERT), and $\mathcal{I}_u$ denotes the set of items interacted with the user u in $\mathcal{D}^{\text{rec}}$:

$$\mathcal{I}_u = \left\{i' : \exists(u, i', c = 1) \in \mathcal{D}^{\text{rec}}\right\}.$$

The corresponding IVs $\mathcal{Z}_{u,i}$ of treatment $\mathcal{T}_{u,i}$ is defined as a set of matrices $\mathbf{Z}_j$:

$$\mathcal{Z}_{u,i} = \{\mathbf{Z}_j : j \in \mathcal{I}_u \cup \{i\}\}, \tag{2}$$

where each matrix $\mathbf{Z}_j$ is defined as a stack of the embeddings of the search queries related to item j. Please note that each $\mathbf{Z}_j$ corresponds to the vector $\mathbf{t}_j$ in treatment $\mathcal{T}_{u,i}$. Specifically, $\mathbf{Z}_j$ can be constructed as follows. First, we retrieve a set of queries from $\mathcal{D}^{\text{src}}$:

$$Q_j = \left\{q : \exists(u', q, j, c = 1) \in \mathcal{D}^{\text{src}}\right\}.$$

After that, the queries in Q_j can be ranked according to, for example, the number of clicks on item j in $\mathcal{D}^{\text{src}}$. The top-N queries are kept, denoted by $\{q_k\}_{k=1}^{N} \subseteq Q_j$. Finally, IVs for the item j, therefore, can be defined as a stack of the embeddings of the top-N queries:

$$\mathbf{Z}_j = \left[\mathbf{t}_{q_1}, \cdots, \mathbf{t}_{q_k}, \cdots, \mathbf{t}_{q_N}\right],$$

where $\mathbf{Z}_j \in \mathbb{R}^{d_q \times N}$, and $\mathbf{t}_{q_k} \in \mathbb{R}^{d_q}$ is the embedding vector of q_k.[1] The query embeddings can be obtained by the model of BERT.

As pre-processing, for each item j in search data $\mathcal{D}^{\text{src}}$, relevant queries Q_j are collected and stacked to compose the IV Z_j offline.

4.3 Treatment reconstruction

Based on the original treatment $\mathcal{T}_{u,i}$ and IVs $\mathcal{Z}_{u,i}$, we show that a new treatment $\mathcal{T}_{u,i}^{\text{re}}$ can be created by first regressing $\mathcal{T}_{u,i}$ on $\mathcal{Z}_{u,i}$ and then combining the fitted vectors and the residuals, shown in right part of Figure 3.

4.3.1 Treatment decomposition.

The goal of IVs method is to isolate the causal association flowing from the treatments to outputs. As shown in Figure 2(b), according to the attributes of IVs (i.e. IVs are unconfounded by the confounder and only affect the outcome Y via treatment X), we regress $\mathcal{T}_{u,i}$ on $\mathcal{Z}_{u,i}$ to get $\widehat{\mathcal{T}}_{u,i}$ which doesn't depend on the confounders B:

$$\widehat{\mathcal{T}}_{u,i} = \left\{\hat{\mathbf{t}}_j = f_{\text{proj}}(\mathbf{t}_j, \mathbf{Z}_j) : j \in \mathcal{I}_u \cup \{i\}\right\}, \tag{3}$$

where $\mathbf{t}_j \in \mathcal{T}_{u,i}$ and $\mathbf{Z}_j \in \mathcal{Z}_{u,i}$ and $f_{\text{proj}} : \mathbb{R}^{d_i} \times \mathbb{R}^{d_q \times N} \mapsto \mathbb{R}^{d_q}$ is defined as a product of matrix $\mathbf{Z}_j$ with an N-dimensional vector τ_j:

$$f_{\text{proj}}(\mathbf{t}_j, \mathbf{Z}_j) = \mathbf{Z}_j \tau_j,$$

where τ_j is a closed form solution of a least square regression:

$$\tau_j = \underset{\tau_j \in \mathbf{R}^N}{\arg\min} \left\|\mathbf{Z}_j \tau_j - \text{MLP}_0(\mathbf{t}_j)\right\|_2^2 = \mathbf{Z}_j^{\dagger} \text{MLP}_0(\mathbf{t}_j),$$

where $\mathbf{Z}_j^{\dagger}$ is the Moore-Penrose pseudoinverse of $\mathbf{Z}_j$ and $\text{MLP}_0 : \mathbb{R}^{d_i} \mapsto \mathbb{R}^{d_q}$ is a multi-layer Perceptron composed of one hidden layer that maps the item j's embedding to a latent space of d_q dimensions. We call $\hat{\mathbf{t}}_j \in \widehat{\mathcal{T}}_{u,i}$ the *fitted part* of the embedding $\mathbf{t}_j$, which reflects the causal association between the embedding and the outcome in RS.

After getting the fitted vectors in $\widehat{\mathcal{T}}_{u,i}$, it is easy to get the *residual part* of the regression $\widetilde{\mathcal{T}}_{u,i}$:

$$\tilde{\mathcal{T}}_{u,i} = \left\{\tilde{\mathbf{t}}_j = \text{MLP}_0(\mathbf{t}_j) - \hat{\mathbf{t}}_j : j \in \mathcal{I}_u \cup \{i\}\right\}, \tag{4}$$

which contains the non-causal association in the RS. The intuition is that the nonlinear representation of the embedding is projected onto the subspace spanned by the columns of the IVs, which separates the fitted part from the residual part. Intervening the fitted part and the residual part differently could help mining the different mechanisms of these two parts for outcome prediction in RS.

Please note that the traditional methods of IVs usually make use of linear models for the regression. Here the linearity assumption is relaxed by first mapping the treatments into a latent space with a nonlinear neural network, which makes our method enjoys both the properties of IVs methods and the powerful representation ability of nonlinear neural networks.

4.3.2 Treatment combination.

The fitted vectors $\widehat{\mathcal{T}}_{u,i}$ and the residuals $\widetilde{\mathcal{T}}_{u,i}$ can be recombined, achieving a reconstructed treatment:

$$\mathcal{T}_{u,i}^{\text{re}} = \left\{\mathbf{t}_j^{\text{re}} = \alpha_j^1 \hat{\mathbf{t}}_j + \alpha_j^2 \tilde{\mathbf{t}}_j : j \in \mathcal{I}_u \cup \{i\}\right\}, \tag{5}$$

where $\hat{\mathbf{t}}_j \in \widehat{\mathcal{T}}_{u,i}$ and $\tilde{\mathbf{t}}_j \in \widetilde{\mathcal{T}}_{u,i}$ are the vectors in these two sets, both correspond to the same item j, and $\alpha_j^1 \in \mathbb{R}$ and $\alpha_j^2 \in \mathbb{R}$ are two combination weights which are estimated by two MLPs:

$$\alpha_j^1 = \text{MLP}_1(\text{MLP}_0(\mathbf{t}_j), \mathbf{Z}_j); \quad \alpha_j^2 = \text{MLP}_2(\text{MLP}_0(\mathbf{t}_j), \mathbf{Z}_j),$$

where the inputs of the two different MLPs are concatenations of the transformed $\mathbf{t}_j$ and $\mathbf{Z}_j$ corresponding to item j.

In traditional causal inference, the major challenge is how to identify the causal association from observed data. Therefore, the residual part is often discarded to remove the effects from confounders. That is, removing the edge from the confounders B to the residual $\tilde{\mathcal{T}}_{u,i}$ in Figure 2(b). In recommendation scenario, however, we still focus on promoting the accuracy of preference estimation, rather than just identifying cause-effects. Existing studies also found that the non-causal associations can contribute to the prediction accuracy [44]. The observation motivates us that not all confounders (biases) need to be discarded. The residual can be leveraged to improve the recommendation performance.

[1] To ensure Z_j is a $d_q \times N$ matrix, we recall several similar query embeddings to be inserted to the right side of $\mathbf{Z}_j$ if $|Q_j| < N$. More details can be found in the experiment.

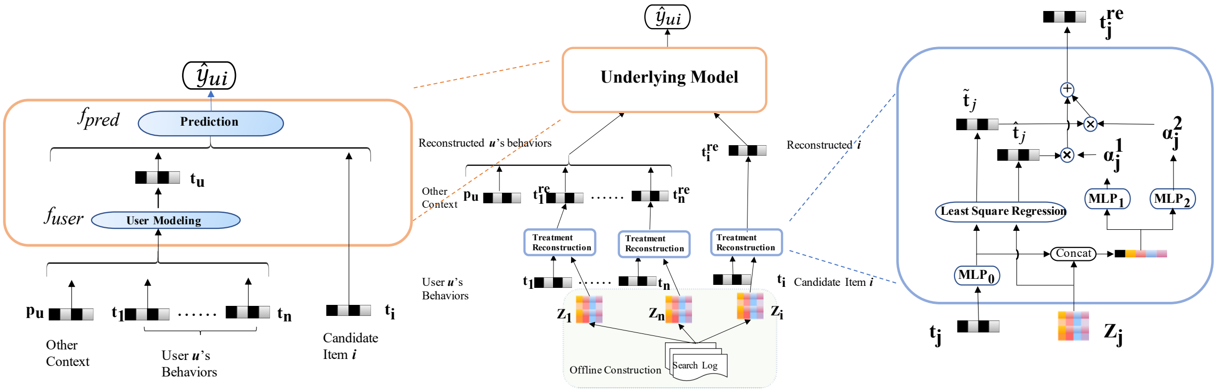


Figure 3: The architecture of IV4Rec framework. Middle: the procedure of IV4Rec applied to the underlying model. Left: structure of the underlying model. Right: detailed implementation of treatment reconstruction.

4.4 Model-agnostic application

Many RSs [31, 37, 46] share a similar structure, which we refer to as the underlying model, shown in the left part of Figure 3. Underlying models represent items as embedding vectors, utilize user's historical behaviors to learn user representation, and predict the preference score of (u, i) based on their learned representations. Our proposed IV4Rec is a model-agnostic framework that can be implemented over existing RSs that follow this underlying structure by simply adding a treatment reconstruction module for item embedding, shown in Figure 3. The procedure follows the causal graph in Figure 2(b), where queries are utilized as IVs to reconstruct the treatments.

Formally, representations of (u, i) in existing RSs, are denoted as $\mathbf{t}_u$ and $\mathbf{t}_i$, where $\mathbf{t}_u$ can be calculated by aggregating the user's historically interacted items and other context information (e.g, user profile, search history and etc). After reconstructing treatments, we can get reconstructed item embedding t_i^{re} and reconstructed user embedding t_u^{re}, where t_u^{re} is calculated as:

$$\mathbf{t}_u^{\text{re}} = f_{\text{user}}(\mathbf{t}_1^{\text{re}}, \mathbf{t}_2^{\text{re}}, \cdots, \mathbf{t}_n^{\text{re}}, \mathbf{p}_u), \tag{6}$$

where $\mathbf{t}_1^{\text{re}}, \mathbf{t}_2^{\text{re}}, \cdots, \mathbf{t}_n^{\text{re}}$ are reconstructed item vectors in the set of interacted items $\mathcal{I}_u$, $\mathbf{p}_u$ is the representation of other context for user u and f_{user} can be any module that learns user representation from user's behaviors, e.g. attention mechanism in [31, 46]. Finally, user's preference is predicted based on learned user/item representation:

$$\hat{y}_{u,i} = f_{\text{pred}}(t_u^{\text{re}}, t_i^{\text{re}}), \tag{7}$$

where f_{pred} can be any model that predicts the preference score from their representations, e.g. MLP [46] or inner product [31].

Please note that the trained treatment reconstruction module can be applied to the items in an offline manner. That is, after the parameters (i.e., the parameters in MLP_0, MLP_1, and MLP_2) in the treatment reconstruction module are determined, the module can be used to reconstruct all of the item embeddings as a pre-processing step. At the online time, the underlying model directly uses the reconstructed items. Therefore, IV4Rec doesn't have any additional time cost at the online recommendation.

4.5 Model training

Parameters in the proposed IV4Rec include parameters in MLP_0, MLP_1, MLP_2, and the parameters from the underlying recommendation model. All these trainable parameters are denoted as $\boldsymbol{\Theta}$ and trained based on $\mathcal{D}^{\text{rec}}$. Specifically, the task of model training amounts to optimizing the following cross entropy loss:

$$\mathcal{L}_{\Theta} = -\frac{1}{|\mathcal{D}^{\text{rec}}|} \sum_{(u,i,c)\in\mathcal{D}^{\text{rec}}} c \cdot \log \hat{y}_{u,i} + (1-c) \cdot \log(1-\hat{y}_{u,i}) + \lambda \|\Theta\|^2, \tag{8}$$

where $\hat{y}_{u,i}$ is the predicted preference socre for (u, i), $\|\Theta\|^2$ is the regularizer term for avoiding over-fitting, and $\lambda > 0$ is a coefficient.

5 DISCUSSION

5.1 Feasibility of using search queries as IVs

According to the theory of IVs estimations, the IVs have two assumptions: exogeneity and relevance.

As for exogeneity, it means that the IVs (search queries) are uncorrelated with the (unobservable) confounders. In recommendation, the common confounders are, for example, variant biases including the position biases, selection biases, etc. Note that currently the search and recommendation are usually deployed as two separate services in one app. The queries are issued when the users are conducting search while the biases occur when the users are accessing the service of RS. Also, these queries may be issued by the users other than the one who is accessing the RS. Therefore, these search users cannot be influenced by the ranking positions/exposure of the items in the RS.

As for relevance, it means that the IVs (search queries) are the causes of the treatment, but do not directly affect the outcomes in RS, i.e., user's click behavior. The search and recommendation share a common goal: providing users with items for satisfying their information needs. In search, the users' information needs are explicitly summarized as queries. In RSs, the information needs are implicitly summarized with the representations of users and items. When the search engine and RS are deployed in one app and serve the same group of users with the same set of items, a large extent of search queries reflect some part of the user information needs in recommendation. The phenomenon indicates that search queries can be seen as a cause of the treatment in recommendation. Considering that the IVs (search queries associated with an item) are specific requests made by users in search, it is obvious that they do not directly affect the outcomes in RS.

Therefore, we conclude that the embeddings of the search queries satisfy the assumptions of exogeneity and relevance well. They are valid IVs for recommendation.

5.2 Difference with traditional IVs methods

In the field of causal inference, IVs methods provide a very powerful framework for learning cause-effects between treatments and outcome even in the presence of confounders. The proposed IV4Rec, inspired by the IVs methods, enjoys a number of merits from IVs, including the elegant approach to involving external search information for constructing IVs for recommendation, the least square regression for decomposing the treatment. However, IV4Rec has made the following fundamental modifications for adapting the traditional method of IVs to recommendation.

(1) **Representation of treatment**: We take the origin embedding as the input of deep neural network and obtain a neural representation of the embedding, rather than directly applying the least square regression. In particular, we update this neural representation by minimizing the loss of the CTR prediction in the recommendation task, making our application of IVs as an end-to-end process. The modification makes the proposed model enjoys the advantages from both IVs and neural networks.

(2) **Reconstruction of treatment with both causal and non-causal parts**: In traditional IVs methods, the residual of the least square regression is discarded. In our approach, however, the residual is used as the embedding representation of the indirect association part. This is because our goal is not just identifying the causal associations. Finding a suitable reconstruction of the causal part and the non-causal part from the original treatment is more helpful in enhancing the recommendation accuracy.

In the recommendation task, biases are ubiquitous, e.g., selection bias and popularity bias, while these biases are usually mixed and difficult to identify. In this paper, we do not model the biases explicitly, and focus on improving the recommendation performance using search data as IVs, which explores the causal relationship between the search and recommendation tasks and eliminates the effects of biases by reconstructing a unified treatment. Since IVs can be used to adjust for both observed and unobserved confounding effects, the proposed model can be considered as a causal learning framework for recommendation using search data.

6 EXPERIMENTS

We present experimental results in this section.[2]

6.1 Experimental settings

6.1.1 Datasets.

IV4Rec requires both search logs and recommendation logs. In the experiments, we created two datasets: one is collected from logs of Kuaishou short-video app, and the other is based on the publicly available MIND dataset [32]. Table 2 shows some of the statistics on both datasets.

Kuaishou Dataset: The Kuaishou dataset is created based on the activities of 12,000 randomly selected users when they elected to use both the search and recommendation services on an app named Kuaishou[3], one of the largest short-video platforms in China, over a period of 7 days in May 2021. The historical behaviors in search and recommendation services of each user were collected. For each user, item and query in the dataset, the user context embedding (64 dimensions), item embedding (64 dimensions), and query embedding (64 dimensions) were generated using existing pre-trained and ranking models from the platform. The detail algorithms are omitted due to privacy concerns.

We split the dataset into three subsets in chronological order, i.e., the first 5 days for training, the 6th day for validation, and the last day for testing. The mini-batch size is set to be 50.

MIND Dataset: To the best of our knowledge, there is no publicly available dataset that contains both user's search and recommendation activities. Therefore, we enhance the MIND[4] [32] data, a benchmark for news recommendation, by generating queries from its metadata. Specifically, motivated by the observation in [25], one search query for each news article was created by concatenating the texts of its category, subcategory and the entities in the metadata. For a few number of articles where the entities are missing, "NLTK"[5] was used to extract entities from the titles. To generate the query and item embeddings, we follow [32] by using BERT [7] to generate the item embeddings (768 dimensions), where the input is the concatenation of the title and abstract. We use the same BERT model to generate query embeddings (768 dimensions) using query strings as input. We directly use users' histories applied in news click histories of the dataset. Users without histories are removed. Users' news click histories are truncated at 50.

Since MIND does not contain a test set with labels, the original training(validation) data is used as training(test) set in the experiments. The mini-batch size is set to be 512.

6.1.2 Baselines and evaluation metrics.

The proposed IV4Rec is model-agnostic, which can be applied to the following baselines and can improve their performances.

NRHUB [31]: NRHUB utilizes an attentive multi-view learning framework for news recommendation to aggregate heterogeneous behaviors of users such as search queries, clicked items, and browsed items. On the experiments of MIND dataset, it was adapted by removing the query encoder module and news encoder module in user representation learning because MIND doesn't support users' search history. On the experiments of Kuaishou dataset, query encoder is removed since items are short-videos other than articles.

DIN [46]: DIN applies an attention mechanism to mine user interests from historical behaviors w.r.t. a certain candidate item. It was adapted to Kuaishou dataset by adding queries and clicked items in the search history as additional history of user behaviors.

We also compare IV4Rec to JSR [41] that jointly optimizes search and recommendation. JSR is a general joint training framework that trains a separate search model and recommendation model by optimizing a joint loss. The search component of JSR was designed as a fully-connected feed-forward network, following original paper.

[2] Codes available at https://github.com/Ethan00Si/Instrumental-variables-for-recommendation.

[3] https://www.kuaishou.com/en

[4] https://msnews.github.io/

[5] https://www.nltk.org/

Table 1: Performance comparisoins of IV4Rec and the baselines on the Kuaishou dataset and the MIND dataset. $*$ and $\dagger$ respectively indicate the improvements over NRHUB and DIN are statistically significant(p−value < 0.05)

Model	Kuaishou Dataset				MIND Dataset			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
NRHUB	0.6455	0.1816	0.4347	0.4692	0.6595	0.3123	0.3428	0.4065
JSR-NRHUB	0.6488	0.1812	0.4326	0.4687	0.6660	0.3164*	0.3480*	0.4117*
IV4Rec-NRHUB	**0.6574***	**0.1837***	**0.4411***	**0.4774***	**0.6722***	**0.3271***	**0.3609***	**0.4219***
DIN	0.6512	0.1833	0.4416	0.4743	0.6851	0.3326	0.3680	0.4304
JSR-DIN	0.6524	0.1838	0.4417	0.4755	0.6873	0.3315	0.3686	0.4308
IV4Rec-DIN	**0.6561**†	**0.1844**	**0.4432**†	**0.4779**†	**0.6898**†	**0.3336**	**0.3700**†	**0.4326**†

Table 2: Statistics of datasets used in this paper.

Dataset	User	Item	Query	Interaction
Kuaishou	12,000	3,053,966	162,624	4,001,613
MIND	736,349	130,380	130,380	95,447,571

The recommendation component was set as NRHUB or DIN, leading to two versions of JSR: **JSR-NRHUB** and **JSR-DIN**.

The proposed IV4Rec is model-agnostic. In the experiments, we applied IV4Rec to the following baselines, achieving two versions of our approach, referred to as **IV4Rec-DIN** and **IV4Rec-NRHUB**.

As for evaluation metrics, AUC was adopted to measure the prediction accuracy on the clicks. MRR and nDCG at the positions of 5, and 10 were also used to measure the accuracy of item rankings, using the clicks as relevance labels. We reported the average results in AUC, MRR, nDCG@5 and nDCG@10 of all impressions.

6.1.3 Implementation details. The hyper-parameters of neural networks were optimized using grid search. The learning rate was selected from $\{1e-4, 3e-4, 5e-4, 7e-4, 1e-3\}$ and the dropout keep probability was selected from $\{0.5, 0.9, 1.0\}$. For the baselines, we set the parameters as the optimal values reported in the original paper. Adam [16] is used to conduct the optimization.

As described in section 4.2, top N associated queries were used to construct the IVs. N was set to 10 on the Kuaishou dataset. In Table 2, the query-click data is very sparse and most items have few associated search clicks. To overcome the sparsity problem, we leveraged cosine similarity of the item embedding and the query embedding to measure the strength of the association. Query-item pairs with high cosine similarity were used as complementary to the sparse query-click data. On the MIND dataset, N was set to 1 since only one query was created for each item (news article).

6.2 Experimental results

From the results reported in Table 1, we found that IV4Rec-NRHUB and IV4Rec-DIN significantly outperformed the corresponding underlying models, NRHUB and DIN, on both datasets, with statistical significance. The results verified the effectiveness of the model-agnostic IV4Rec framework in improving any recommendation models. On the other hand, IV4Rec-NRHUB and IV4Rec-DIN also outperformed the baselines of JSR-NRHUB and JSR-DIN, which jointly optimize search and recommendation. Please note that NRHUB leveraged search activities for user modeling and DIN was adapted by adding users' search histories on the Kuaishou dataset. Thus the improvements achieved on the Kuaishou dataset were attributed to IV4Rec instead of adding users' search history features. The results verified the effects of using search queries as IV for reconstructing treatments in recommendation.

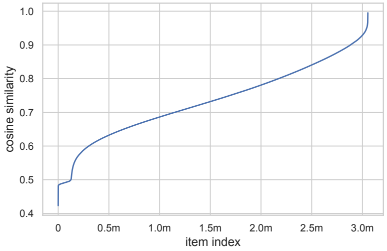


Figure 4: Distribution of cosine similarity between each item and its highest ranked query. The items are sorted by cosine similarity from the lowest to the highest. Each x-axis index refers to a unique item.

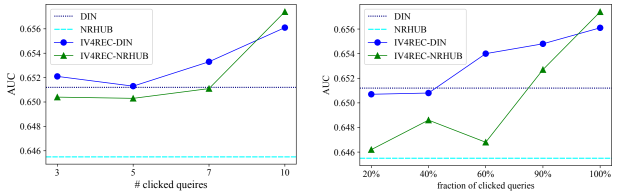
6.3 Detailed empirical analysis

We conducted more detailed experiments to show how and why IV4Rec can improve the recommendation accuracy.

6.3.1 Effects of search queries as IVs. To verify the relevance assumption in Section 5.1, experiments were conducted on the Kuaishou dataset. Specifically, we tested the relevance between items and their corresponding queries. As discussed in 6.1.3, we used cosine similarity of query-item pair embeddings to measure the relevance. The embeddings of queries and items were generated using pretrained models from the platform. The similarity of each item and its highest ranked query was plotted in Figure 4. From the results, most similarity scores were higher than 0.6, indicating that most items were highly relevant with corresponding queries.

We explored the impacts of the number of relevant queries per item. Specifically, we tested the performances of IV4Rec when the number of queries extracted for each item (i.e., N in $\mathbf{Z}_j \in \mathbb{R}^{d_q \times N}$) as the IVs. Figure 5(a) showed the AUC curves of IV4Rec models w.r.t. $N = 3, 5, 7, 10$. We found that with the increased number of N (more related queries means higher relevance between IV and treatment), AUC also increased for both IV4Rec-NRHUB and IV4Rec-DIN.

We also tested the performances of IV4Rec when a few queries in the IVs were selected randomly rather than the high-ranked queries according to the clicks in search. Figure 5(b) illustrates the AUC curves w.r.t. 20%, 40%, 60%, 80% and 100% of the queries are clicked queries (others are random queries). From the results, we can see



(a) AUC w.r.t. # clicked queries (b) AUC w.r.t. fraction of clicked queries

Figure 5: AUC curves when the IVs (search queries) are selected differently on Kuaishou dataset. Two horizontal lines denotes the performances of DIN and NRHUB, respectively. IV4REC significantly outperforms the baselines when each item has more than 5 relevant queries.

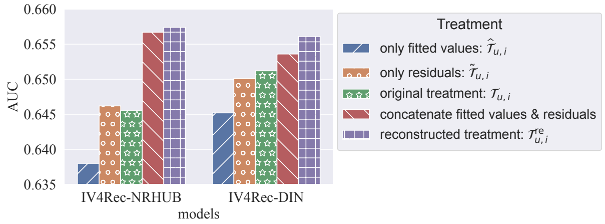
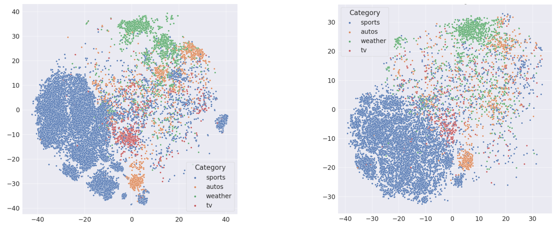


Figure 6: Impact of different treatment reconstruction methods w.r.t. AUC on the Kuaishou dataset.

that when more queries in IVs were set randomly (lower relevance of IVs to treatments), more hurts to the performances of IV4Rec-NRHUB and IV4Rec-DIN. Based on the results, we conclude that the clicked search queries are effective IVs for recommendation.

6.3.2 Effects of using residuals in recommendation. In traditional IV estimation (e.g., 2SLS Kmenta [17]), the residuals are discarded. In IV4Rec, we utilized both the fitted part and the residual part. Experiments were done to test the AUC of different modified IV4Rec versions on the Kuaishou dataset. They are using only the fitted values $\hat{\mathcal{T}}_{u,i}$, using only the residuals $\tilde{\mathcal{T}}_{u,i}$, using the original treatment $\mathcal{T}_{u,i}$ without reconstruction, using the reconstructed treatment by concatenating $\hat{\mathcal{T}}_{u,i}$ and $\tilde{\mathcal{T}}_{u,i}$, using $\mathcal{T}^{re}_{u,i}$ reconstructed by IV4Rec. From the results shown in Figure 6, we can see that both the fitted part and the residual part have contributions in recommendation. The AUC improved a lot when these two parts are combined together as a reconstructed treatment. The phenomenon can be observed when both NRHUB and DIN were used as the underlying model of IV4Rec. The results indicate that though they represent the non-causal associations, the residual part can still contribute to the user preference prediction. The reason is that the residuals still have a strong association with the outcome. When the goal is making accurate prediction rather than analyzing the causal effects, the fitted part and the residual part are complementary.

Compared to the two versions of combination, the proposed IV4Rec, which uses weighted combination and two MLPs to estimate the weights, performed better than the simple concatenation. The results verified the effectiveness of the treatment reconstruction method in Section 4.3.



(a) Original embeddings by BERT (b) Reconstructed embeddings in IV4Rec-NRHUB

Figure 7: Visualization of the item embeddings on MIND dataset. Using IV4Rec can better cluster embeddings within the same category.

6.3.3 Enhancing the item embeddings. We conducted experiments to illustrate whether the reconstructed treatments are better item embeddings than the original ones. The experiments were conducted based on MIND dataset because each news article in MIND falls into a category. We selected the articles from four categories (sports, autos, weather, and tv), and illustrated their original embeddings $\mathbf{t}_j$'s (by BERT) in Figure 7(a) with t-SNE [28] where the four colors indicate four categories. Based on IV4Rec-NRHUB, we also calculated the reconstructed item embeddings $\mathbf{t}^{re}_j$'s of these news articles, and illustrated them in Figure 7(b). Comparing these two figures, we found that the reconstructed embeddings are distributed better than the original embeddings. For example, the 'sports' articles are more tightly clustered at the bottom-left corner. The results indicate that IV4Rec has the ability to improve the item embeddings with the help of search queries. It also provides an explanation of why IV4Rec can improve the underlying model.

7 CONCLUSIONS

In this paper, we proposed a model agnostic IV-based causal learning framework to improve recommendation using search data, called IV4Rec. IV4Rec made use of the search queries as IVs and decomposed the recommendation embeddings into the causal association part and the non-causal association part, mining the different mechanisms of these two parts for preference prediction in recommendation. Besides, IV4Rec combined the traditional method of instrumental variables with deep neural networks and provided an end-to-end framework for estimating the model parameters. Experiments on Kuaishou product data and a public benchmark demonstrated the effectiveness of IV4Rec in recommendation.

ACKNOWLEDGMENTS

This work was funded by the National Key R&D Program of China (2019YFE0198200), Kuaishou, the National Natural Science Foundation of China (61872338, 62006234, 61832017), Project funded by China Postdoctoral Science Foundation (No. 2021T140722), Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, Intelligent Social Governance Interdisciplinary Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China, and Public Policy and Decision-making Research Lab of Renmin University of China.

REFERENCES

[1] Belloni A, Chen D, Chernozhukov V, and Hansen C. 2012. Sparse models and methods for optimal instruments with an application to eminent domain. *Econometrica* 80, 6 (2012), 2369–2429.

[2] Aman Agarwal, Ivan Zaitsev, Xuanhui Wang, Cheng Li, Marc Najork, and Thorsten Joachims. 2019. Estimating Position Bias without Intrusive Interventions. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*, J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman (Eds.). 474–482.

[3] Stephen Bonner and Flavian Vasile. 2018. Causal embeddings for recommendation. In *Proceedings of the 12th ACM conference on recommender systems*. 104–112.

[4] Mehmet Caner and Bruce E Hansen. 2004. Instrumental variable estimation of a threshold model. *Econometric Theory* 20, 5 (2004), 813–843.

[5] V. Chernozhukov, G. W. Imbens, and W. K. Newey. 2007. Instrumental variable estimation of nonseparable models. *Journal of Econometrics* 139, 1 (2007), 4–14.

[6] W. Bruce Croft, Donald Metzler, and Trevor Strohman. 2009. *Search Engines: Information Retrieval in Practice.* Pearson Education.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, Jill Burstein, Christy Doran, and Thamar Solorio (Eds.). Association for Computational Linguistics, 4171–4186.

[8] Hector Garcia-Molina, Georgia Koutrika, and Aditya Parameswaran. 2011. Information seeking: convergence of search, recommendations, and advertising. *Commun. ACM* 54, 11 (2011), 121–130.

[9] Sahin Cem Geyik, Stuart Ambler, and Krishnaram Kenthapadi. 2019. Fairness-aware ranking in search & recommendation systems with application to linkedin talent search. In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining*. 2221–2231.

[10] Jon Atle Gulla, Lemei Zhang, Peng Liu, Özlem Özgöbek, and Xiaomeng Su. 2017. The adressa dataset for news recommendation. In *Proceedings of the international conference on web intelligence*. 1042–1048.

[11] Jason Hartford, Greg Lewis, Kevin Leyton-Brown, and Matt Taddy. 2017. Deep IV: A flexible approach for counterfactual prediction. In *International Conference on Machine Learning*. PMLR, 1414–1423.

[12] Jason S. Hartford, Greg Lewis, Kevin Leyton-Brown, and Matt Taddy. 2017. Deep IV: A Flexible Approach for Counterfactual Prediction. In *Proceedings of the 34th International Conference on Machine Learning*. 1414–1423.

[13] José Miguel Hernández-Lobato, Neil Houlsby, and Zoubin Ghahramani. 2014. Probabilistic matrix factorization with non-random missing data. In *International Conference on Machine Learning*. PMLR, 1512–1520.

[14] Guido W Imbens and Donald B Rubin. 2015. *Causal inference in statistics, social, and biomedical sciences.* Cambridge University Press.

[15] Rashidul Islam, Kamrun Naher Keya, Ziqian Zeng, Shimei Pan, and James Foulds. 2021. Debiasing career recommendations with neural fair collaborative filtering. In *Proceedings of the Web Conference 2021*. 3779–3790.

[16] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[17] Jan Kmenta. 2010. Mostly harmless econometrics: An empiricist's companion.

[18] Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. 2017. Counterfactual Fairness. In *Advances in Neural Information Processing Systems 30*. 4066–4076.

[19] Dawen Liang, Laurent Charlin, and David M Blei. 2016. Causal inference for recommendation. In *Causation: Foundation to Application, Workshop at UAI. AUAI*.

[20] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling Fairness and Bias in Dynamic Learning-to-Rank. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 429–438.

[21] Yusuke Narita, Shota Yasui, and Kohei Yata. 2021. Debiased Off-Policy Evaluation for Recommendation Systems. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 372–379.

[22] Zohreh Ovaisi, Ragib Ahsan, Yifan Zhang, Kathryn Vasilaky, and Elena Zheleva. 2020. Correcting for Selection Bias in Learning-to-rank Systems. In *Proceedings of the Web Conference 2020*. 1863–1873.

[23] Judea Pearl. 2009. *Causality: Models, Reasoning, and Inference.* Cambridge university press.

[24] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor. 2011. *Recommender Systems Handbook.* Springer.

[25] Jennifer Rowley. 2000. Product search in e-sProduct search in e-shopping: a review and research propositionshopping: a review and research propositions. *Journal of Consumer Marketing* 17, 1 (2000), 20–35.

[26] 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.

[27] Xiaoyuan Su and Taghi M Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in artificial intelligence* 2009 (2009).

[28] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[29] Arun Venkatraman, Wen Sun, Martial Hebert, J Andrew Bagnell, and Byron Boots. 2016. Online Instrumental Variable Regression with Applications to Online Linear System Identification. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence.*

[30] Yixin Wang, Dawen Liang, Laurent Charlin, and David M Blei. 2020. Causal inference for recommender systems. In *Fourteenth ACM Conference on Recommender Systems*. 426–431.

[31] Chuhan Wu, Fangzhao Wu, Mingxiao An, Tao Qi, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with heterogeneous user behavior. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 4874–4883.

[32] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3597–3606.

[33] Le Wu, Lei Chen, Pengyang Shao, Richang Hong, Xiting Wang, and Meng Wang. 2021. Learning Fair Representations for Recommendation: A Graph-based Perspective. In *Proceedings of the Web Conference 2021*. 2198–2208.

[34] Tao Wu, Ellie Ka-In Chio, Heng-Tze Cheng, Yu Du, Steffen Rendle, Dima Kuzmin, Ritesh Agarwal, Li Zhang, John Anderson, Sarvjeet Singh, et al. 2020. Zero-Shot Heterogeneous Transfer Learning from Recommender Systems to Cold-Start Search Retrieval. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2821–2828.

[35] Jun Xu, Xiangnan He, and Hang Li. 2018. Deep learning for matching in search and recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1365–1368.

[36] Liyuan Xu, Yutian Chen, Siddarth Srinivasan, Nando de Freitas, Arnaud Doucet, and Arthur Gretton. 2021. Learning Deep Features in Instrumental Variable Regression. In *International Conference on Learning Representations*.

[37] Feng Xue, Xiangnan He, Xiang Wang, Jiandong Xu, Kai Liu, and Richang Hong. 2019. Deep item-based collaborative filtering for top-n recommendation. *ACM Transactions on Information Systems (TOIS)* 37, 3 (2019), 1–25.

[38] Jing Yao, Zhicheng Dou, Ruobing Xie, Yanxiong Lu, Zhiping Wang, and Ji-Rong Wen. 2021. USER: A Unified Information Search and Recommendation Model based on Integrated Behavior Sequence. *arXiv preprint arXiv:2109.15012* (2021).

[39] Jiawei Yao, Jiajun Yao, Rui Yang, and Zhenyu Chen. 2012. Product recommendation based on search keywords. In *2012 Ninth Web Information Systems and Applications Conference*. IEEE, 67–70.

[40] Junkun Yuan, Anpeng Wu, Kun Kuang, Bo Li, Runze Wu, Fei Wu, and Lanfen Lin. 2022. Auto IV: Counterfactual Prediction via Automatic Instrumental Variable Decomposition. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16, 4 (2022), 1–20.

[41] Hamed Zamani and W. Bruce Croft. 2018. Joint Modeling and Optimization of Search and Recommendation. In *Proceedings of the First Biennial Conference on Design of Experimental Search & Information Retrieval Systems, Bertinoro, Italy, August 28-31, 2018 (CEUR Workshop Proceedings, Vol. 2167)*, Omar Alonso and Gianmaria Silvello (Eds.). CEUR-WS.org, 36–41.

[42] Hamed Zamani and W Bruce Croft. 2020. Learning a joint search and recommendation model from user-item interactions. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 717–725.

[43] 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.

[44] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal Intervention for Leveraging Popularity Bias in Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 11–20.

[45] Yu Zheng, Chen Gao, Xiang Li, Xiangnan He, Yong Li, and Depeng Jin. 2021. Disentangling User Interest and Conformity for Recommendation with Causal Embedding. In *Proceedings of the Web Conference 2021*. 2980–2991.

[46] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1059–1068.