Causal Factorization Machine for Robust Recommendation

Yunqi Li, Hanxiong Chen, Juntao Tan, Yongfeng Zhang
Department of Computer Science, Rutgers University, New Brunswick, NJ, US
{yunqi.li,hanxiong.chen,juntao.tan,yongfeng.zhang}@rutgers.edu

ABSTRACT

Factorization Machines (FMs) are widely used for the collaborative recommendation because of their effectiveness and flexibility in feature interaction modeling. Previous FM-based works have claimed the importance of selecting useful features since incorporating unnecessary features will introduce noise and reduce the recommendation performance. However, previous feature selection algorithms for FMs are proposed based on the i.i.d. hypothesis and select features according to their importance to the predictive accuracy on training data. However, the i.i.d. assumption is often violated in real-world applications, and shifts between training and testing sets may exist. In this paper, we consider achieving causal feature selection in FMs so as to enhance the robustness of recommendation when the distributions of training data and testing data are different. What's more, different from other machine learning tasks like image classification, which usually select a global set of causal features for a predictive model, we emphasize the importance of considering personalized causal feature selection in recommendation scenarios since the causal features for different users may be different. To achieve our goal, we propose a personalized feature selection method for FMs and refer to the confounder balancing approach to balance the confounders for every treatment feature. We conduct experiments on three real-world datasets and compare our method with both representative shallow and deep FM-based baselines to show the effectiveness of our method in enhancing the robustness of recommendations and improving the recommendation accuracy.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

KEYWORDS

Recommender Systems; Factorization Machine; Robustness; Personalization; Causal Inference; Confounder Balancing.

ACM Reference Format:
Yunqi Li, Hanxiong Chen, Juntao Tan, Yongfeng Zhang. 2022. Causal Factorization Machine for Robust Recommendation. In *The ACM/IEEE Joint Conference on Digital Libraries in 2022 (JCDL '22), June 20–24, 2022, Cologne, Germany*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3529372.3530921

JCDL '22, June 20–24, 2022, Cologne, Germany

ACM ISBN 978-1-4503-9345-4/22/06…$15.00
https://doi.org/10.1145/3529372.3530921

1 INTRODUCTION

Factorization machines (FMs) are originally proposed for the collaborative recommendation, and are widely used in predictive analytics such as targeted advertising [15] and toxicogenomics prediction [53]. To leverage the interactions between features, FMs provide a generic way to model second-order feature interactions to enhance the linear regression model. Specifically, FMs associate a weight for each feature or feature interaction, and predicts the target through the weighted sum of all features. Some previous works have emphasized that it is necessary to perform feature selection for FMs to effectively filter out useless feature interactions, since incorporating unnecessary feature interactions will introduce noise and degrade the recommendation performance [8, 9]. However, previous FMs-based feature selection algorithms are proposed based on the i.i.d. hypothesis, i.e., they assume the testing data is drawn independently from the same distribution as the training data, so that the model learned from the training data can be directly applied to the testing data and still achieves satisfactory performance. Therefore, previous works select features according to their importance to the accuracy of the predictive model on training data. However, in real applications, especially in recommender systems, the i.i.d. hypothesis is usually violated. For example, agnostic selection bias may occur when collecting data, so that the distributions of training and testing data may be different [26, 43]. Moreover, in real applications, the trained recommender systems will be used to make recommendations for new users or items, whose data is not even available during the training process. Therefore, how to select features that are always important among different domains so as to achieve robust recommendation which performs consistently well on even non-i.i.d. data is of paramount importance and necessity. In this paper, we focus on achieving robust factorization machines for recommendation tasks.

It is well recognized that a reasonable way to achieve a robust predictive model is to learn the model with causal features, whose effects on the target variable is insensitive to the shifts among different domains [20, 39]. Generally, there are two different types of input features for learning a predictive model. One is called causal features, which are the features that have a causal structural relationship with the prediction target. Another type of feature is known as the noisy features, which do not have a causal relationship with the target variable, but may be correlated with the causal features or the target variable. The noisy features will not have any causal effect on the target variable conditioned on all the causal features. Therefore, the conditional distribution of the target variable given the causal features keeps invariant across different domains. In this paper, we consider achieving a robust factorization machine by selecting causal features to make predictions.

What's more, different from other machine learning tasks such as image classification, which usually only need to select a common set of causal features for all samples, the causal feature selection

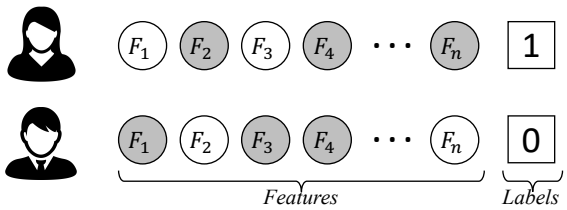

Figure 1: Example of personalized causal feature selection of different users. The shaded features are the selected causal features, which could be different for different users.

of FMs for recommendation tasks should be personalized. This is because each user will have personalized preference on features and thus the causal features of each user can be different, as shown in Figure.1. Just as an example, for one user, the causal feature for predicting whether he or she likes a coat may be the coat's price and comfort, while for another user, whether the coat is fashionable and popular may be more important. Therefore, it is important to consider the personalized preferences of different users when achieving causal feature selection. The main challenge of detecting the personalized causal feature of each user is that we usually have extremely sparse data for training in recommendation scenarios. And the limited interactions from each user make it difficult to select user-specific features and feature interactions. To this end, we learn personalized user embeddings to capture user preferences, and factorize the coefficients in FM by the product of learned user embeddings and feature embeddings so that we not only can reduce the total number of parameters to save computing and storage costs, but also can leverage the advantage of collaborative filtering.

To select personalized causal features for each user, we need to estimate the causal effect of each feature on the target variable. However, the feature space of FMs is usually huge since FMs can incorporate rich sources of information associated with users or items such as user interactions, the demographic features of users, descriptions of items, as well as the context information and sequential dependencies, etc. What's more, FMs take feature interactions into consideration, which also increases the dimensionality of the feature space. Moreover, there is usually no prior knowledge about which first- or second-order features have a causal relationship with the target variable and which features are noisy features. Therefore, it is infeasible to intervene in each and every first- or second-order feature to identify its causal effect. Under such considerations, we take advantage of the idea of confounder balancing [20, 39] to identify the causal effect of features on the target variable by balancing the distributions of confounders across different treatment features. The confounder balancing approach usually learns a weight matrix to reweight each sample so that FMs can naturally assign a weight to each first- or second-order feature that can imply the causal effect of each feature on the target variable, and thus help to select causal features for achieving robust factorization machine. There are several causal inference based methods that are proposed to balance the distributions of confounders across different treatment levels for identifying the causal effect of features, including the Markov blankets methods [18, 33], propensity score reweighting methods [3, 4], and the confounder balancing approach [2, 12], etc. In this paper, we borrow the idea of the confounder balancing approach since it is the most suitable to handle the settings of FMs which usually contain large feature space and sparse input data.

In this paper, we focus on personalized causal feature selection for FMs to achieve robust recommendations. The proposed method will help to establish more accurate and stable recommender systems to better serve the diverse users in various areas such as e-commerce, social networks, and digital libraries. The key contributions of this paper are as follows:

- We enhance the robustness of FMs for recommendation under the non-i.i.d. setting where the distributions of training and testing data may be different due to some agnostic bias.
- To achieve our goal, we select causal features for FMs since their effects on the target variable are insensitive to the shifts among different domains. Besides, we emphasize that the causal features selected for recommendation should be personalized to satisfy users' different preferences.
- Technically, we propose a personalized feature selection method for factorization machine, and refer to the confounder balancing approach to push the coefficients of FMs towards the causal effect of each feature on the target variable.
- We conduct experiments on three real-world datasets with both shallow and deep FM-based baselines to show the effectiveness of our method in improving recommendation accuracy and enhancing robustness in the recommendation.

In the following, we review related work in Section 2. In Section 3, we introduce the details of our proposed method Causal Factorization Machine (CFM). Experimental settings and results are provided in Section 4. Finally, we conclude this work in Section 5.

2 RELATED WORK

2.1 Factorization Machine

Factorization Machines (FMs) [35, 36] are popular supervised learning models which combine the advantages of factorization models [19] and Support Vector Machines (SVMs) [40]. Due to their great ability in modeling feature interactions, FMs have been widely used for feature-based collaborative recommendation.

In recent years, many FM variants have been proposed and achieved success in recommendation scenarios [11, 13, 31, 47]. Examples include Field-Aware Factorization Machine [15] which associates multiple embedded vectors of a feature to distinguish its interaction with other features of different fields; Sparse Factorization Machine [32] which aims to learn the sparse feature interactions; as well as Higher-Order Factorization Machine [5] which trains FMs with higher-order interactions by an efficient algorithm; Neural Factorization Machine [13] which combines the linearity of FM in modeling second-order feature interactions and the non-linearity of neural networks in modeling higher-order feature interactions; Attentional Factorization Machine [46] which leverages an attention network to learn the importance of each feature interaction; Boosted Factorization Machines [54] which includes contextual information into FMs for the context-aware recommendation, etc.

2.2 Feature Selection for Factorization Machine

As FMs can incorporate generous auxiliary data of users or items for training, and take second-order feature interactions into consideration, the dimensionality of the feature space of FMs is relatively high. It has been shown that not all feature interactions are useful, and incorporating unnecessary feature interactions may introduce noise and degrade recommendation performance [8, 9]. Therefore, besides the many variants of FMs, some recent works also pay attention to the feature interaction selection problem of FMs. Cheng et al. [9] select feature interaction based on gradient boosting. Xu et al. [49] propose an efficient interaction selection approach via sparse FMs by applying group Lasso [61] to feature embeddings. Mao et al. [29] select context features for context-aware recommendation with FM based on the predictive power of features. Chen et al. [8] select personalized feature interactions by a Bayesian variable selection approach with spike and slab priors. Liu et al. [23] propose a two-stage automatic feature interaction selection approach which can automatically identify important feature interactions for factorization models.

In this work, we consider selecting causal features for FMs to achieve robust recommendations. Our work is different from previous feature selection algorithms of FMs in the following aspects: 1) Previous works are proposed under the i.i.d. hypothesis, which is often impossible to achieve in reality, while our work tries to improve the generalizability of FMs when selection bias between training and testing data may exist. 2) The motivation of previous feature selection works for FMs is to reduce the dimensionality of the FM feature space so as to save the computing cost as well as improve the recommendation accuracy by reducing noise, while our work aims to enhance the robustness of recommendation by selecting causal features. 3) Some previous works take prior knowledge to select features, e.g., in [8], the authors assume that if some first-order features are not selected for the FM, then the second-order interaction of their combinations will neither be selected, so that they can prune lots of useless features. In contrast, we treat every feature the same since there is usually no prior knowledge about which feature is more important. The causal features will be naturally selected during FM training by obtaining weights that imply their causal effect on the target once we balance the confounders in the predictive model. 4) Previous works usually only consider selecting second-order features or non-personalized features, instead, we consider selecting personalized causal first- and second-order features for users, which can better capture users' preferences over different features.

2.3 Robust Factorization Machine

Previous works which aim to enhance the robustness of FMs mainly focus on the model's defensive ability against attacks to improve the adversarial robustness of the model. However, robustness is a multi-dimensional concept and may have different requirements for recommender systems. For example, the robustness can be evaluated based on the performance variance of the model under the i.i.d. hypothesis or the non-i.i.d. generalizability of the model [45]. FM models against adversarial perturbations have been studied recently [24, 34]. Punjabi and Bhatt [34] model the perturbation when there is data uncertainty through Gaussian or Poisson perturbations on the input signals. Liu et al. [24] consider the situation where there are noisy training samples by making some labels of the input features wrong. Liu et al. [27] consider discrete adversarial perturbation on instance features since the considered FM features are binary. In our work, we focus on improving the model generalizability of FMs when there exist distributional shifts between training and testing data since the i.i.d. hypothesis is easily violated in real applications.

2.4 Causal Inference in Recommendation

Recently, the research community has explored causal learning to improve recommender systems from several different perspectives such as: 1) Improving the explainability of recommender systems, e.g., Ghazimatin et al. [10] generate provider-side counterfactual explanations by finding a minimal set of user's historical actions which if removed can lead to a change of the recommendation; Tan et al. [41, 42] and Xu et al. [51] propose counterfactual explainable recommendation which is able to formulate the complexity and strength of explanations, and seek simple and effective explanations for the model decision. 2) Improving the fairness of recommender systems:, e.g., Li et al. [22] achieve personalized counterfactual fairness in the recommendation for users. 3) Mitigating data bias in recommender systems: e.g., Schnabel et al. [38] provide a principled framework to handle selection biases by adapting models and estimation techniques from causal inference. Liu et al. [25] and Zhang et al. [57] solve the bias problems in recommendation based on counterfactual learning to enable uniform data modeling. Xu et al. [50, 52] propose causal collaborative filtering and provide a general framework for modeling counterfactual reasoning in the recommendation. 4) Improving recommendation performance, e.g., Wang et al. [44] and Xiong et al. [48] proposed causal data augmentation based on counterfactual reasoning for recommendation. In this work, we aim to achieve robust FMs by selecting causal features.

3 CAUSAL FACTORIZATION MACHINE

In this section, we introduce the Causal Factorization Machine (CFM) proposed for the robust recommendation. We first introduce factorization machines (FMs) and how we can capture the personalized preference of users in the framework of FMs. After that, we introduce how to select personalized causal features of FMs based on observational data through balancing confounders.

3.1 Factorization Machine

Factorization machines [35, 36] are generic supervised learning models which were originally proposed for the feature-based collaborative recommendation. In recommendation task, FMs predict users' preferences on items based on their feature vectors $\boldsymbol{x} \in \mathbb{R}^n$, where $\boldsymbol{x}$ is a given real-valued feature vector with n features. FMs estimate the target variable by modeling all features as well as the second-order interactions between each pair of features. A general formulation for FMs is shown below:

$$\hat{y}(\boldsymbol{x}) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij} x_i x_j \tag{1}$$

where w_0 models the global bias; w_i models the first-order feature and denotes the weight of the i-th feature for estimating the target; and w_{ij} models the second-order feature interaction between feature i and j, and denotes the weight of the cross feature $x_i x_j$ for estimating the target. $x_i \in \boldsymbol{x}$ is the i-th feature of $\boldsymbol{x}$ and $\hat{y}(\boldsymbol{x})$ is the predicted rating for $\boldsymbol{x}$. The learned $\hat{y}(\boldsymbol{x})$ can be applied to a variety of prediction tasks such as regression tasks, classification tasks as well as ranking and recommendation tasks. To learn the weight of second-order features, FMs factorize it as $w_{ij} = \langle \mathbf{v}_i, \mathbf{v}_j \rangle = \mathbf{v}_i^T \mathbf{v}_j$, rather than learning each individual interaction parameter w_{ij}, where $\mathbf{v}_i \in \mathbb{R}^d$ is the embedding of feature i with dimension d. This is the key point to estimate high quality weight parameters for second-order interactions under sparse data [35].

Eq.(1) shows that FMs associate a weight for each first- and second-order feature, and predict the target through the weighted sum of all features. The weights of features are learned based on their correlation with the target variable in the training data. However, recent works have shown that the spurious correlation between noisy features and target variable is a major cause of the accuracy drop of current models under distribution shifts [1, 21, 28, 56]. The spurious correlations are intrinsically caused by the subtle correlations between noisy features and causal features. Therefore, to enhance the robustness of FMs, it is necessary to eliminate the spurious correlation by selecting causal features. In this work, we do not incorporate additional parameters for causal feature selection, but require the weight of each first- and second-order feature to imply its causal effect on estimating the target variable, so that the causal features can be directly selected through obtaining higher weights and thus help to enhance the robustness of FMs.

3.2 Personalized Factorization Machine

To select causal features in recommendation scenario, we need to consider the personalized preference of different users for features. However, the feature weights in FMs are global, which means that FMs consider the effect of each feature on the target variable as the same for different users. Although we see that if we take user ID as input features, the FM in Eq.(1) can capture personalization by involving a bias for each user, it still fails to personalize second-order features. Besides, only considering a bias parameter for each user is not enough for estimating personalized causal effect of each feature on the target variable. To achieve our goal, we reformulate Eq.(1) as a personalized FM by introducing personalized feature parameters as the following:

$$\hat{y}_u(\boldsymbol{x}) = w_u + \sum_{i=1}^{n} w_{ui} x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{uij} x_i x_j \tag{2}$$

where w_u is the personalized bias of user u; w_{ui} and w_{uij} reflect the preferences of user u over first- and second-order feature interactions. However, it is infeasible to directly estimate the parameters in Eq.(2) since we usually need to face extreme sparse settings in recommendation task, and there is usually not enough data to achieve high quality estimation for personalized coefficients. Therefore, inspired by the factorization models, we break the independence of the parameters by factorizing them through user embeddings and feature embeddings. By factorizing the parameters, we not only can reduce the total number of parameters to save computing and storage costs, but also can leverage the advantage of collaborative filtering since the data for one interaction can also help to estimate the parameters for related interactions. We formulate the personalized FMs by Eq.(3):

$$\hat{y}_u(\boldsymbol{x}) = w_u + \sum_{i=1}^{n} \langle \mathbf{u}, \mathbf{v}_i \rangle x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle \mathbf{u}, \mathbf{v}_i \odot \mathbf{v}_j \rangle x_i x_j \tag{3}$$

Here, we learn a user embedding $\mathbf{u} \in \mathbb{R}^d$ for each user to capture user preferences, and a feature embedding $\mathbf{v} \in \mathbb{R}^d$ for each feature to capture feature characters. The w_{ui} is factorized by the product of user embedding and feature embedding of i-th feature; while w_{uij} is factorized by the product of user embedding as well as the feature embeddings of the i- and j-th features. Here, $\mathbf{v}_i \odot \mathbf{v}_j$ is the Hadamard product of the feature embeddings $\mathbf{v}_i$ and $\mathbf{v}_j$. Now the coefficients in FMs can reflect the user's personalized preference. The next problem is how to make the weight of each feature represent its causal impact on the target variable so as to achieve causal feature selection.

3.3 Causal Feature Selection

To identify causal features, the most important problem is how to estimate their causal effects on the target variable. It is well known that conducting randomized experiments is the gold standard for recognizing the causal effect of a variable. However, it is usually impractical to achieve randomized experiments in real cases due to its expensive cost and the hurt to user experience. Therefore, it is necessary to estimate the causal effect of the input features on the target variable directly from observational data. However, the idea of randomized experiments is still instructive. In randomized experiments, the causal effect of treatment variables can be easily estimated since all the confounding variables have been controlled. Therefore, in this work, we refer to the confounder balancing approach [2, 12, 20, 39] for estimating the causal effect of features based on observational data.

The techniques of confounder balancing are usually used to estimate the causal effect of variables based on only observational data. In observational studies, the treatment is usually not randomly assigned due to some agnostic bias, and the distributions of covariates, i.e., all the variables affecting both the treatment and the outcome, are different between treatment and control groups. Therefore, to precisely estimate causal effects under such a setting, the distributions of confounders between treatment and control groups need to be balanced to correct agnostic bias. To balance the distributions of confounders, most of the confounder balancing methods choose to directly balance the moments of confounders since moments can uniquely determine a distribution, and balancing the moments between treatment and control groups can be conducted by adjusting sample weights $\boldsymbol{W}$ as follows [2, 12]:

$$\begin{aligned} \boldsymbol{W} &= \arg\min_{\boldsymbol{W}} \| M_{T=1}(\boldsymbol{W}) - M_{T=0}(\boldsymbol{W}) \|_2^2 \\ &= \arg\min_{\boldsymbol{W}} \left\| \frac{\sum_{i:T_i=1} W_i \cdot \mathbf{X}_i}{\sum_{i:T_i=1} W_i} - \frac{\sum_{i:T_i=0} W_i \cdot \mathbf{X}_i}{\sum_{i:T_i=0} W_i} \right\|_2^2 \end{aligned} \tag{4}$$

Here, $\boldsymbol{W}$ is the weights for sample reweighting. T is a treatment variable. $M_{T=1}(\boldsymbol{W})$ represents the first-order moments of variables

$\mathbf{X}$ on treatment ($T = 1$) group, while $M_{T=0}(\boldsymbol{W})$ represent the first-order moments of variables $\boldsymbol{X}$ on control ($T = 0$) group. Once we balance the distributions of confounders and reweight samples by $\boldsymbol{W}$ learned from Eq.(4), the correlation between a treatment and outcome variable can now imply the causal effect of the treatment on the outcome variable, and we can take the average difference of the outcome variable Y between treatment and control groups as the causal effect of treatment variable T on Y.

As there is usually no prior knowledge of the causal structure of input variables in FMs, we have to treat every feature as a treatment variable and estimate its causal effect on the outcome. When we treat one feature as treatment, the other features are considered as confounders [12]. In this paper, we take the unconfoundedness assumption [37] that there are no unobserved confounders, i.e., all the covariates affecting both the treatment and the outcome are observed and can be controlled. And the distribution of treatment and outcome are independent when conditioned on the observed variables. Therefore, we can estimate the causal effect of each feature from observational data by adequately controlling all covariates.

Specifically, suppose the input matrix for FMs is $\boldsymbol{X}$, where every row of $\boldsymbol{X}$ is a training sample, and every column of $\boldsymbol{X}$ is a feature. To identify the causal contribution of the j-th feature $\boldsymbol{X}_{\cdot,j}$, we treat the feature $\boldsymbol{X}_{\cdot,j}$ as a treatment variable, the label $\boldsymbol{Y}$ as the outcome variable, and all the remaining features $\boldsymbol{X}_{\cdot,-j} = \boldsymbol{X} \backslash \boldsymbol{X}_{\cdot,j}$ as confounders. To identify the causal effect of a treatment feature, we need to remove the confounding bias between the treatment and control groups induced by all the confounders $\boldsymbol{X}_{\cdot,-j}$, so that the correlation between $\boldsymbol{X}_{\cdot,j}$ and label $\boldsymbol{Y}$ represents the causal effect of $\boldsymbol{X}_{\cdot,j}$ on $\boldsymbol{Y}$, and the causal features are naturally selected by obtaining higher weights during the learning process.

3.4 Learning Objective

To select causal features for FMs, we need to take every first- and second-order feature as treatments. Therefore, for estimating the causal effects of all features, we need to learn a global sample weight that balances the distributions of confounders between treatment and control group for every treatment feature [20, 39]. In this work, without loss of generality, we consider all the features as binary since the features of FMs are usually discrete [27], and continuous or categorical features can be easily converted to binary features by one-hot encoding. Therefore, we can assign each sample to either treatment group or control group according to the value of the treatment feature, i.e., if the treatment feature takes the value of 1 in a sample, then the sample is a treatment sample, otherwise, it is a control sample. The global sample weight is learned by the following loss:

$$\boldsymbol{W} = \arg\min_{\boldsymbol{W}} \sum_{i=1}^{n} \sum_{j=i}^{n} \left\| \frac{\boldsymbol{X}_{\cdot,-ij}^{T} \cdot (\boldsymbol{W} \odot \boldsymbol{X}_{\cdot,ij})}{\boldsymbol{W}^{T} \cdot \boldsymbol{X}_{\cdot,ij}} - \frac{\boldsymbol{X}_{\cdot,-ij}^{T} \cdot (\boldsymbol{W} \odot (1 - \boldsymbol{X}_{\cdot,ij}))}{\boldsymbol{W}^{T} \cdot (1 - \boldsymbol{X}_{\cdot,ij})} \right\|_2^2 \tag{5}$$

where $\boldsymbol{W} \in \mathbb{R}^{m \times 1}$ is the global sample weights for balancing confounders. $\boldsymbol{X}_{\cdot,i}$ is the i-th feature, i.e. the i-th column of $\boldsymbol{X}$, and $\boldsymbol{X}_{\cdot,ij}$ refers to the second-order feature interacted by the i-th and j-th features, i.e., $\boldsymbol{X}_{\cdot,ij} = \boldsymbol{X}_{\cdot,i} \cdot \boldsymbol{X}_{\cdot,j}$. Note that as features are binary, $\boldsymbol{X}_{\cdot,i} = \boldsymbol{X}_{\cdot,ii} = \boldsymbol{X}_{\cdot,i} \cdot \boldsymbol{X}_{\cdot,i}$. Therefore, the first-order features are also considered into Eq.(5). $\boldsymbol{X}_{\cdot,-ij} = \boldsymbol{X} \backslash \boldsymbol{X}_{\cdot,ij}$ represents all the remaining features except for $\boldsymbol{X}_{\cdot,ij}$. Eq.(5) represents the total loss of confounder balancing when setting every feature as treatment variable.

FMs can be applied to a variety of prediction tasks including regression, classification and ranking with different objective functions. In this work, we simply take the square loss as follows to predict whether a user will interact with an item, and optimize the objective function by stochastic gradient descent:

$$L = \sum_{\boldsymbol{x} \in \mathcal{T}} \left(\hat{y}(\boldsymbol{x}) - y(\boldsymbol{x}) \right)^2 \tag{6}$$

where $\mathcal{T}$ denotes the set of training instances. For the learning objective of CFM, we integrate the confounder balancing regularizer and the square loss to jointly optimize the sample weights $\boldsymbol{W}$ and the personalized FMs coefficients $\boldsymbol{\theta} = \{w_u, w_{ui}, w_{uij}\}$:

$$\begin{aligned} \min \quad & \textstyle\sum_{k=1}^{m} W_k \cdot \left(\hat{y}_{\boldsymbol{\theta}}(\boldsymbol{X}_{k,\cdot}) - y_{\boldsymbol{\theta}}(\boldsymbol{X}_{k,\cdot}) \right)^2 \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{n} \sum_{j=i}^{n} \left\| \frac{\boldsymbol{X}_{\cdot,-ij}^{T} \cdot (\boldsymbol{W} \odot \boldsymbol{X}_{\cdot,ij})}{\boldsymbol{W}^{T} \cdot \boldsymbol{X}_{\cdot,ij}} - \frac{\boldsymbol{X}_{\cdot,-ij}^{T} \cdot (\boldsymbol{W} \odot (1 - \boldsymbol{X}_{\cdot,ij}))}{\boldsymbol{W}^{T} \cdot (1 - \boldsymbol{X}_{\cdot,ij})} \right\|_2^2 \leq \epsilon_1 \\ & \|\boldsymbol{\theta}\|_2^2 \leq \epsilon_2, \boldsymbol{W} \geq 0, \|\boldsymbol{W}\|_2^2 \leq \epsilon_3, \left(\textstyle\sum_{k=1}^{m} \boldsymbol{W}_k - 1 \right)^2 \leq \epsilon_4 \end{aligned} \tag{7}$$

where m is the number of samples, n is the number of first-order features, $\boldsymbol{X}_{k,\cdot}$ is the k-th row of the input matrix $\boldsymbol{X}$. The norm $\|\boldsymbol{\theta}\|_2^2 \leq \epsilon_2$ is to avoid overfitting. The term $\boldsymbol{W} \geq 0$ requires each of the sample weights to be non-negative. The norm $\|\boldsymbol{W}\|_2^2 \leq \epsilon_3$ is to reduce the variance of the sample weights. The last constraint $\left(\sum_{k=1}^{m} \boldsymbol{W}_k - 1 \right)^2 \leq \epsilon_4$ aims to guarantee that the model will not converge to the naive solution where all the sample weights are learned to be zero [39].

4 EXPERIMENT

In this section, we conduct experiments to answer the following research questions:

- **RQ1:** Can CFM boost the performance of factorization machines based on personalized causal features selection?
- **RQ2:** Can we enhance the robustness of factorization machines based on CFM under the non-i.i.d. setting when there are agnostic bias between training and testing data?
- **RQ3:** Is it necessary to consider personalized preference of users when selecting causal features in recommendation?

In the following subsections, we will first introduce the experimental settings such as the data we use and the baseline models for performance comparisons. After that, we will provide detailed analysis to the experimental results.

4.1 Dataset

We evaluate the proposed model based on three publicly available datasets. Each dataset contains user-item interactions together with features on both user and item side. For quality and efficiency consideration, we sample denser and smaller datasets from the original data for our experiments.

RentTheRunWay[1]. This dataset is introduce in [30]. It comes from a website that allows women to rent clothes for various purposes. It contains features such as user ratings, product categories,

[1] https://cseweb.ucsd.edu/ jmcauley/datasets.html

catalog sizes, users' measurements, etc. It contains 832 users, 1,299 items and 10,974 interactions.

Post[2]. This is a Kaggle dataset which includes the posts that each user has viewed in social network for post recommendation. The goal is to predict the posts that each user will potentially interact with in the future. It contains user-side features such as gender and academics (undergraduate or graduate), which are two binary features. On item-side it contains item category information which is a 20-class feature. It contains 468 users, 1,371 items and 22,882 interactions.

MovieLens[3]. This is a movie recommendation dataset which contains user-item interactions, user ratings, user profile information as well as movie features. Each user has gender, age and occupation as the user features, where gender is a binary feature, occupation is a 21-class feature, and for age, users are assigned into 7 groups based on their age range. Each movie has 20 genres, which are 20 binary features. The year information of the movies is mapped into 18 categories. It contains 809 users, 920 movies and 16,941 interactions.

The range of user ratings in the RentTheRunWay dataset is from 2 to 10. Following [30], we convert the user ratings into binaries by mapping the ratings that are greater or equal than eight to 1 and that are lower or equal than seven as well as those unobserved interactions to 0. The range of user ratings in the MovieLens dataset is from 1 to 5. Similarly, we map the ratings greater or equal than three to 1 and the rest to 0. The Post dataset contains implicit feedback, which means that we only have 1's in the dataset. To train the models, we couple each positive user-post pair with one sampled negative pair. This negative sample is extracted from the posts that the user has never interacted before [13]. We conduct leave-one-out [7, 16] for train-validate-test construction, i.e., we randomly sample one interaction from each user to create the test set, sample another one for the validation set, and the remaining interactions are used as the training set.

4.2 Baselines

To evaluate the effectiveness of our proposed model, we select several representative factorization machine models for comparison.

- **FM** [35]: The Factorization Machine (FM) model, which leverages second-order feature interactions for prediction.
- **DeepFM** [11]: This work ensembles deep neural network for feature learning and the power of factorization machine for recommendation tasks.
- **FNFM** [55] Field-aware Neural Factorization Machines, which uses deep neural network to capture higher-order feature combinations.
- **AFM** [46]: Attentional Factorization Machine, which extends FM by using attention mechanism to distinguish the importance of second-order cross features. AFM incorporates additional attentive weights for selecting useful feature interactions based on i.i.d. assumption.

[2]https://www.kaggle.com/vatsalparsaniya/post-pecommendation
[3]https://grouplens.org/datasets/movielens/

4.3 Evaluation Methods

We evaluate the performance of the models under the Top-K recommendation tasks. We consider standard metrics including Normalized Discounted Cumulative Gain (NDCG@K) and the Hit ratio (Hit@K) scores to evaluate the recommendation quality [14]. Both of the two metrics are the higher the better. In the following, we take the abbreviations N@K and H@K to represent NDCG@K and Hit@K respectively. Since computing the user-item pairwise scores for each user over the entire item space for ranking is quite inefficient, we conduct negative sampling for evaluation instead [58]. For each user, we randomly select 100 negative samples that the user has never interacted with. These negative items are put together with the positive item in the validation or test set to constitute the user's candidates list [6, 7, 58]. Then the metric scores are computed over this candidates list to evaluate the recommendation model's Top-K ranking performance. The result of all metrics in our experiments are averaged over all users. To ensure the reliability, all metric scores in the result tables are the average of ten random experiments.

4.4 Experimental Settings

To reasonably evaluate the model performance, we tune the hyper-parameters of all the models to get their best performance based on the validation set. For our personalized causal feature selection model CFM, we set the size of user embedding and feature embedding to 64. For FM and AFM, the feature embedding and attention hidden vector dimensions are also set to 64. For DeepFM, the embedding and hidden state dimensions are set to 64, which gives its best performance. The drop rate for AFM and DeepFM is set to 0.2. For FNFM, the embedding size is set to 4 and the neural network hidden size is 64.

We set the hyper-parameters in Eq.(7) as follows: the ϵ_1, which is for achieving confounder balancing, is set to 0.1; to avoid overfitting, we apply ℓ_2 regularization on the parameters of CFM and set ϵ_2 to 10^{-4}. and the ℓ_2 regularization is also applied to all baseline models with the same hyper-parameter; the ϵ_3 and ϵ_4, which are for reducing the variance of the sample weights and avoiding the naive solutions, are set to 1. The learning rate is set to 0.001. We conduct batch training by setting the size of each batch to 256. We apply Adam [17] as the optimization algorithm to update the model parameters.

4.5 RQ1: Recommendation Accuracy of CFM

First, we study the question of whether CFM can achieve better recommendation accuracy than the baseline models under the i.i.d. setting. We directly test the model performance on the testing set which is generated by a random leave-one-out setting. The experimental results of all the models on three datasets are shown as the overall results in Table.1-3, respectively. From the results, we can see that CFM achieves the best Hit Ratio and NDCG on all three datasets except for the H@5 on MovieLens. However, we can see later that although FNFM achieves better H@5 on MovieLens, it is less robust on this dataset than CFM. The results show that it is effective to consider personalized causal feature selection in CFM even under the i.i.d. setting.

Table 1: The recommendation performance of our CFM method and baselines on the overall testing data, as well as multiple subgroups with different selection bias on the *RentTheRunWay* dataset. The subgroups are selected according to different selection bias with respect to the age of users. The results are reported in percentage (%). The evaluation metrics here are calculated based on the top-10 predictions in the test set. The best results are highlighted in bold.

	FM				DeepFM				FNFM				AFM				CFM			
	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5
Overall	2.67	4.29	1.85	2.52	2.97	4.80	2.10	2.85	2.70	4.86	1.94	2.82	2.55	4.41	1.76	2.51	**4.14**	**6.65**	**3.15**	**4.17**
G1	2.65	3.96	1.81	2.35	3.61	5.65	2.48	3.30	2.65	4.65	1.89	2.71	2.04	3.87	1.44	2.18	4.87	7.26	3.73	4.71
G2	2.46	4.24	1.72	2.46	2.76	4.44	1.99	2.69	2.58	4.74	1.89	2.78	2.61	4.44	1.77	2.52	3.77	6.33	2.84	3.88
G3	3.12	4.78	2.16	2.84	2.67	4.52	1.89	2.65	3.02	5.33	2.10	3.03	3.02	4.98	2.10	2.89	4.02	6.59	3.09	4.13
DR	37.09	29.52	39.50	32.24	29.84	16.76	32.28	23.26	28.28	17.86	26.83	17.68	36.69	22.51	35.71	24.44	**15.65**	**11.50**	**18.51**	**13.95**

Table 2: The recommendation performance of our CFM method and baselines on the overall testing data, as well as multiple subgroups with different selection bias on the *Post* dataset. The subgroups are selected according to different selection bias with respect to the academic level of users. The results are reported in percentage (%). The evaluation metrics here are calculated based on the top-10 predictions in the test set. The best results are highlighted in bold.

	FM				DeepFM				FNFM				AFM				CFM			
	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5
Overall	2.48	3.91	1.78	2.38	2.74	4.79	1.93	2.78	2.91	4.91	2.01	2.83	2.80	4.66	1.97	2.73	**3.70**	**6.24**	**2.71**	**3.74**
G1	2.60	3.48	1.97	2.33	2.47	4.45	1.67	2.49	2.55	4.10	1.78	2.41	2.60	4.36	1.82	2.54	3.52	5.46	2.63	3.39
G2	2.36	4.31	1.61	2.52	2.99	5.10	2.18	3.06	3.24	5.68	2.21	3.22	2.99	4.94	2.11	2.90	3.86	6.97	2.78	4.07
DR	29.05	21.41	29.81	19.89	21.46	24.15	24.57	25.64	21.21	18.57	22.82	19.39	21.95	**17.99**	25.11	19.49	**16.12**	18.79	**13.19**	**15.73**

Table 3: The recommendation performance of our CFM method and baselines on the overall testing data, as well as multiple subgroups with different selection bias on the *MovieLens* dataset. The subgroups are selected according to different selection bias with respect to the age of users. The results are reported in percentage (%). The evaluation metrics here are calculated based on the top-10 predictions in the test set. The best results are highlighted in bold.

	FM				DeepFM				FNFM				AFM				CFM			
	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5	H@3	H@5	N@3	N@5
Overall	3.68	6.13	2.67	3.66	4.09	7.09	2.92	4.26	5.68	**9.22**	4.08	5.52	3.41	5.15	2.50	3.21	**6.16**	9.13	**4.66**	**5.91**
G1	3.84	6.72	2.86	4.05	3.65	6.72	2.58	3.89	5.61	9.17	4.04	5.48	3.28	5.30	2.49	3.32	6.64	9.60	5.02	6.22
G2	3.56	5.64	2.45	3.29	4.50	7.78	3.22	4.65	4.82	8.32	3.45	4.88	3.02	4.63	2.23	2.88	6.64	9.20	5.12	6.18
G3	3.46	5.67	2.50	3.40	4.38	7.11	3.20	4.37	6.17	9.71	4.43	5.89	3.73	5.23	2.65	3.26	5.39	8.60	4.03	5.33
DR	29.04	20.24	29.80	24.23	27.22	18.36	27.60	20.44	22.36	19.22	24.71	21.23	42.69	28.87	44.64	32.90	**21.30**	**14.39**	**23.24**	**18.59**

4.6 RQ2: Robustness of CFM

Next, we study whether CFM can enhance the robustness of FMs for the Top-K recommendation task. In this paper, we consider improving the generalizability of the model when there is selection bias between the training and testing sets. Here in our paper, the selection bias between training and testing data refers to the agnostic bias when collecting data so that the distributions of training and testing sets are different [20, 39]. To create distributional shifts between the training and testing data, we select multiple subsets from the testing set according to different selection biases with respect to different personal features of the users. In the RentTheRunWay dataset, we split the testing data into 3 subsets according to users' age, including Group 1 (G1): Age $\in [0, 30)$, Group 2 (G2): Age $\in [30, 40)$ and Group 3 (G3): Age $\in [40, \infty)$. In the Post dataset, we split the testing data into 2 subsets according to users' academic level, including Group 1 (G1): Undergraduate and Group 2 (G2): Graduate. In the MovieLens dataset, we also split the testing data into 3 subsets according to users' age. Different from the RentTheRunWay dataset where we can directly group users by age, in the MovieLens dataset, the age groups are already created by the original data, and the age of users are given by a 7-class categorical feature. We group the first three categories as Group 1 (G1) which contains users of the Age $\in [0, 35)$, the fourth and fifth categories as Group 2 (G2) which contains users of the Age $\in [35, 50)$, and the last two categories as Group 3 (G3) which contains users of the Age $\in [50, \infty)$. We test the performance of the models not only on the overall testing set but also on each of these subsets which contain different selection biases.

To evaluate the robustness of models, we refer to the drop rate introduced in [45]. The drop rate (DR) evaluates the drop rate between the recommendation performance P_N on the non-i.i.d. test set (which is usually lower) and the recommendation performance

Table 4: The recommendation performance of FM, balanced FM (B-FM), and our CFM method on the overall testing data, as well as multiple subgroups with different selection bias with respect to user age on *RentTheRunWay* dataset. The results are reported in percentage (%). The evaluation metrics here are calculated based on the top-10 predictions in the test set. The best results are highlighted in bold.

Model	Group	H@3	H@5	N@3	N@5
FM	Overall	2.67	4.29	1.85	2.52
	G1	2.65	3.96	1.81	2.35
	G2	2.46	4.24	1.72	2.46
	G3	3.12	4.78	2.16	2.84
	DR	37.09	29.52	39.50	32.24
B-FM	Overall	2.76	4.50	1.93	2.64
	G1	2.69	4.00	1.82	2.35
	G2	2.63	4.54	1.87	2.65
	G3	3.12	4.98	2.18	2.94
	DR	24.10	17.85	26.75	19.39
CFM	Overall	**4.14**	**6.65**	**3.15**	**4.17**
	G1	4.87	7.26	3.73	4.71
	G2	3.77	6.33	2.84	3.88
	G3	4.02	6.59	3.09	4.13
	DR	**15.65**	**11.50**	**18.51**	**13.95**

P_I on the i.i.d. test set (which is usually higher). This is defined as:

$$DR = \frac{P_I - P_N}{P_I}$$

where the recommendation performance P can be any effectiveness evaluation metric such as Hit Ratio and NDCG. Here, since we have multiple subsets with different selection biases, we show the drop rate under the worst case, i.e., we calculate the drop rate of the model based on the overall testing performance and the worst performance on the subsets as the following:

$$DR = \frac{P_{\text{Overall}} - \min_{G \in \text{Groups}}(P_G)}{P_{\text{Overall}}}$$

We test the model performance of baselines and CFM on all the testing sets, compute the drop rate of all the evaluation metrics and show the results on three datasets in Table.1, Table.2, and Table.3, respectively. To ensure reliability, all metric scores including drop rates in the tables are the average of ten random experiments. We can see that, in general, the baseline models have a significant drop rate which shows the lack of robustness of current representative FM models. For the performance of CFM, on the RentTheRunWay and MovieLens datasets, CFM achieves the smallest drop rate on all the evaluation metrics. For the Post dataset, CFM achieves the best drop rate on most of the evaluation metrics except for H@5. Although FNFM and AFM achieve a lower drop rate on H@5, we can see that CFM achieves better H@5 than FNFM and AFM on not only the overall testing set but also on G1 and G2. This is because the causal weights of CFM are more insensitive to the distributional shifts induced by selection bias, while correlation-based methods tend to be more unstable under such scenarios. In summary, we can see that CFM is effective for enhancing the robustness of FMs for the Top-K recommendation task when there are distributional shifts between the training and testing data.

4.7 RQ3: Effect of Personalized Consideration

To study the necessity of taking users' personalized preferences into consideration when selecting causal features, we directly balance the confounders for training FM (which does not consider personalized feature selection) and get the results of balanced FM (B-FM) in Table.4. Intuitively, B-FM still conducts causal feature selection, but in a global way, i.e., the selected causal features for all users are the same. We can see that by comparing the results of FM, B-FM, and CFM, the CFM model achieves the best overall performance and robustness. Moreover, we can see that although it is not as good as CFM, directly balancing confounders on FM can help to enhance the robustness of FM, however, it is not effective to improve the recommendation performance as the recommendation accuracy of FM and B-FM are very close. This is because directly balancing the confounders on FM is actually trying to select the global causal features for all users. However, since users have personalized preferences on features, the causal effect of each feature on the target label may be different for each user. As a result, forcefully selecting global causal features will not help to improve the performance since the learned model is not suitable for any user but just trying to accommodate the majority. Therefore, it is necessary to consider the personalized preferences of users when selecting causal features for FMs.

5 CONCLUSIONS

In this paper, we focus on improving the robustness of factorization machines for recommendation tasks under the non-i.i.d. setting where distributional shifts may exist between the training and testing set due to some selection bias. To this end, we select causal features for FMs since the effects of causal features on the target variable are insensitive to the shifts across different domains. Furthermore, we show that the causal features selected for the recommendation task should be personalized to satisfy users' different preferences, which is different from other machine learning tasks such as image classification which select a global set of causal features for a predictive model. Therefore, we introduce a personalized causal feature selection method for FMs. We first propose a personalized factorization machine by incorporating personalized coefficients for capturing users' personalized preferences on features. To achieve high-quality estimation for personalized coefficients, we factorize them through user embeddings and feature embeddings to take advantage of collaborative learning. After that, we refer to confounder balancing to balance the confounders for every treatment feature, so that the learned weight of each feature in FMs represents its causal effect on the target variable. We conduct experiments on three real-world datasets and compare our method with both shallow and deep FM-based models to show the effectiveness of our method in enhancing the robustness of recommendations as well as improving the recommendation accuracy. The limitation of the work lies in the efficiency of the constrained-based optimization method when the number of features is huge. In the future work, we will work to propose a more efficient method for causal feature selection in the recommendation.

REFERENCES

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893* (2019).

[2] Susan Athey, Guido W Imbens, and Stefan Wager. 2018. Approximate residual balancing: debiased inference of average treatment effects in high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 80, 4 (2018), 597–623.

[3] Peter C Austin. 2011. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate behavioral research* 46, 3 (2011), 399–424.

[4] Heejung Bang and James M Robins. 2005. Doubly robust estimation in missing data and causal inference models. *Biometrics* 61, 4 (2005), 962–973.

[5] Mathieu Blondel, Akinori Fujino, Naonori Ueda, and Masakazu Ishihata. 2016. Higher-order factorization machines. In *NIPS*. 3351–3359.

[6] Hanxiong Chen, Yunqi Li, Shaoyun Shi, Shuchang Liu, He Zhu, and Yongfeng Zhang. 2022. Graph collaborative reasoning. In *WSDM*. 75–84.

[7] Hanxiong Chen, Shaoyun Shi, Yunqi Li, and Yongfeng Zhang. 2021. Neural Collaborative Reasoning. In *WWW*. 1516–1527.

[8] Yifan Chen, Pengjie Ren, Yang Wang, and Maarten de Rijke. 2019. Bayesian personalized feature interaction selection for factorization machines. In *Proceedings of the 42nd International ACM SIGIR*. 665–674.

[9] Chen Cheng, Fen Xia, Tong Zhang, Irwin King, and Michael R Lyu. 2014. Gradient boosting factorization machines. In *RecSys*. 265–272.

[10] Azin Ghazimatin, Oana Balalau, Rishiraj Saha Roy, and Gerhard Weikum. 2020. PRINCE: Provider-side interpretability with counterfactual explanations in recommender systems. In *Proceedings of the 13th WSDM*. 196–204.

[11] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *IJCAI*. 1725–1731.

[12] Jens Hainmueller. 2012. Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. *Political analysis* 20, 1 (2012), 25–46.

[13] Xiangnan He and Tat-Seng Chua. 2017. Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th International ACM SIGIR*. 355–364.

[14] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *TOIS* 20, 4 (2002), 422–446.

[15] Yuchin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. 2016. Field-aware factorization machines for CTR prediction. In *RecSys*. 43–50.

[16] Santosh Kabbur, Xia Ning, and George Karypis. 2013. Fism: factored item similarity models for top-n recommender systems. In *SIGKDD*. 659–667.

[17] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[18] Daphne Koller and Mehran Sahami. 1996. *Toward optimal feature selection*. Technical Report. Stanford InfoLab.

[19] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD*. 426–434.

[20] Kun Kuang, Peng Cui, Susan Athey, Ruoxuan Xiong, and Bo Li. 2018. Stable prediction across unknown environments. In *SIGKDD*. 1617–1626.

[21] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. 2017. Building machines that learn and think like people. *Behavioral and brain sciences* 40 (2017).

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

[23] Bin Liu, Chenxu Zhu, Guilin Li, Weinan Zhang, Jincai Lai, Ruiming Tang, Xiuqiang He, Zhenguo Li, and Yong Yu. 2020. Autofis: Automatic feature interaction selection in factorization models for click-through rate prediction. In *Proceedings of the 26th ACM SIGKDD*. 2636–2645.

[24] Chenghao Liu, Teng Zhang, Jundong Li, Jianwen Yin, Peilin Zhao, Jianling Sun, and Steven CH Hoi. 2019. Robust Factorization Machine: A Doubly Capped Norms Minimization. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 738–746.

[25] Dugang Liu, Pengxiang Cheng, Zhenhua Dong, Xiuqiang He, Weike Pan, and Zhong Ming. 2020. A general knowledge distillation framework for counterfactual recommendation via uniform data. In *SIGIR*. 831–840.

[26] Haochen Liu, Da Tang, Ji Yang, Xiangyu Zhao, Jiliang Tang, and Youlong Cheng. 2021. Self-supervised Learning for Alleviating Selection Bias in Recommendation Systems. (2021).

[27] Yang Liu, Xianzhuo Xia, Liang Chen, Xiangnan He, Carl Yang, and Zibin Zheng. 2020. Certifiable robustness to discrete adversarial perturbations for factorization machines. In *Proceedings of the 43rd International ACM SIGIR*. 419–428.

[28] David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Scholkopf, and Léon Bottou. 2017. Discovering causal signals in images. In *CVPR*. 6979–6987.

[29] Xueyu Mao, Saayan Mitra, and Viswanathan Swaminathan. 2017. Feature selection for FM-based context-aware recommendation systems. In *ISM*. IEEE, 252–255.

[30] Rishabh Misra, Mengting Wan, and Julian McAuley. 2018. Decomposing fit semantics for product size recommendation in metric spaces. In *RecSys*. 422–426.

[31] Trung V Nguyen, Alexandros Karatzoglou, and Linas Baltrunas. 2014. Gaussian process factorization machines for context-aware recommendations. In *SIGIR*. 63–72.

[32] Zhen Pan, Enhong Chen, Qi Liu, Tong Xu, Haiping Ma, and Hongjie Lin. 2016. Sparse factorization machines for click-through rate prediction. In *ICDM*. IEEE, 400–409.

[33] Jean-Philippe Pellet and André Elisseeff. 2008. Using Markov blankets for causal structure learning. *Journal of Machine Learning Research* 9, 7 (2008).

[34] Surabhi Punjabi and Priyanka Bhatt. 2018. Robust factorization machines for user response prediction. In *Proceedings of the 2018 World Wide Web Conference*. 669–678.

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

[36] Steffen Rendle. 2012. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 3 (2012), 1–22.

[37] Paul R Rosenbaum and Donald B Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70, 1 (1983), 41–55.

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

[39] Zheyan Shen, Peng Cui, Kun Kuang, Bo Li, and Peixuan Chen. 2018. Causally regularized learning with agnostic data selection bias. In *Proceedings of the 26th ACM international conference on Multimedia*. 411–419.

[40] Ingo Steinwart and Andreas Christmann. 2008. *Support vector machines*. Springer Science & Business Media.

[41] Juntao Tan, Shijie Geng, Zuohui Fu, Yingqiang Ge, Shuyuan Xu, Yunqi Li, and Yongfeng Zhang. 2022. Learning and Evaluating Graph Neural Network Explanations based on Counterfactual and Factual Reasoning. In *Proceedings of the ACM Web Conference 2022*. 1018–1027.

[42] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. *CIKM* (2021).

[43] Claire Vernade and Olivier Cappé. 2015. Learning from missing data using selection bias in movie recommendation. In *DSAA*. IEEE, 1–9.

[44] Zhenlei Wang, Jingsen Zhang, Hongteng Xu, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Counterfactual data-augmented sequential recommendation. In *SIGIR*. 347–356.

[45] Chen Wu, Ruqing Zhang, Jiafeng Guo, Yixing Fan, and Xueqi Cheng. 2021. Are Neural Ranking Models Robust? *arXiv preprint arXiv:2108.05018* (2021).

[46] 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 *IJCAI*. 3119–3125.

[47] Fenfang Xie, Liang Chen, Yongjian Ye, Zibin Zheng, and Xiaola Lin. 2018. Factorization machine based service recommendation on heterogeneous information networks. In *ICWS*. IEEE, 115–122.

[48] Kun Xiong, Wenwen Ye, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, Binbin Hu, Zhiqiang Zhang, and Jun Zhou. 2021. Counterfactual Review-based Recommendation. In *CIKM*. 2231–2240.

[49] Jianpeng Xu, Kaixiang Lin, Pang-Ning Tan, and Jiayu Zhou. 2016. Synergies that matter: Efficient interaction selection via sparse factorization machine. In *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM, 108–116.

[50] Shuyuan Xu, Yingqiang Ge, Yunqi Li, Zuohui Fu, Xu Chen, and Yongfeng Zhang. 2021. Causal Collaborative Filtering. *arXiv preprint arXiv:2102.01868* (2021).

[51] Shuyuan Xu, Yunqi Li, Shuchang Liu, Zuohui Fu, Yingqiang Ge, Xu Chen, and Yongfeng Zhang. 2021. Learning causal explanations for recommendation. In *The 1st International Workshop on Causality in Search and Recommendation*.

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

[53] Makoto Yamada, Wenzhao Lian, Amit Goyal, Jianhui Chen, Kishan Wimalawarne, Suleiman A Khan, Samuel Kaski, Hiroshi Mamitsuka, and Yi Chang. 2017. Convex factorization machine for toxicogenomics prediction. In *KDD*. 1215–1224.

[54] Fajie Yuan, Guibing Guo, Joemon M Jose, Long Chen, Haitao Yu, and Weinan Zhang. 2017. Boostfm: Boosted factorization machines for top-n feature-based recommendation. In *IUI*. 45–54.

[55] Li Zhang, Weichen Shen, Jianhang Huang, Shijian Li, and Gang Pan. 2019. Field-aware neural factorization machine for click-through rate prediction. *IEEE Access* 7 (2019), 75032–75040.

[56] Xingxuan Zhang, Peng Cui, Renzhe Xu, Linjun Zhou, Yue He, and Zheyan Shen. 2021. Deep Stable Learning for Out-Of-Distribution Generalization. In *CVPR*. 5372–5382.

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

[58] Wayne Xin Zhao, Junhua Chen, Pengfei Wang, Qi Gu, and Ji-Rong Wen. 2020. Revisiting Alternative Experimental Settings for Evaluating Top-N Item Recommendation Algorithms. In *CIKM*. 2329–2332.