The Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24)

CoreRec: A Counterfactual Correlation Inference for Next Set Recommendation

Kexin Li[1], Chengjiang Long[2*], Shengyu Zhang[1], Xudong Tang[1], Zhichao Zhai[1], Kun Kuang[1], Jun Xiao[1]

[1] Zhejiang University
[2] Meta Reality Labs
{12221004,sy_zhang,22051028, wheltz,junx}@zju.edu.cn, cjfykx@gmail.com, kkun2010@gmail.com

Abstract

The next set recommendation aims to predict the items that are likely to be bought in the next purchase. Central to this endeavor is the task of capturing intra-set and cross-set correlations among items. However, the modeling of cross-set correlations poses challenges due to specific issues. Primarily, these correlations are often implicit, and the prevailing approach of establishing an indiscriminate link across the entire set of objects neglects factors like purchase frequency and correlations between purchased items. Such hastily formed connections across sets introduce substantial noise. Additionally, the preeminence of high-frequency items in numerous sets could potentially overshadow and distort correlation modeling with respect to low-frequency items. Thus, we devoted to mitigating misleading inter-set correlations. With a fresh perspective rooted in causality, we delve into the question of whether correlations between a particular item and items from other sets should be relied upon for item representation learning and set prediction. Technically, we introduce the Counterfactual Correlation Inference framework for next set recommendation, denoted as CoreRec. This framework establishes a counterfactual scenario in which the recommendation model impedes cross-set correlations to generate intervened predictions. By contrasting these intervened predictions with the original ones, we gauge the causal impact of inter-set neighbors on set prediction—essentially assessing whether they contribute to spurious correlations. During testing, we introduce a post-trained switch module that selects between set-aware item representations derived from either the original or the counterfactual scenarios. To validate our approach, we extensively experiment using three real-world datasets, affirming both the effectiveness of CoreRec and the cogency of our analytical approach.

Introduction

In retail and e-commerce contexts, it is customary for patrons to make multi-item acquisitions within a single transaction, often termed a 'purchase set.' Such sets acquired sequentially may unveil inherent interdependencies. Consequently, it becomes instinctive to discern users' inclinations and fathom their underlying motives by scrutinizing their historical purchase sets. This scrutiny can then pave the way for anticipatory forecasting of ensuing purchase sets.

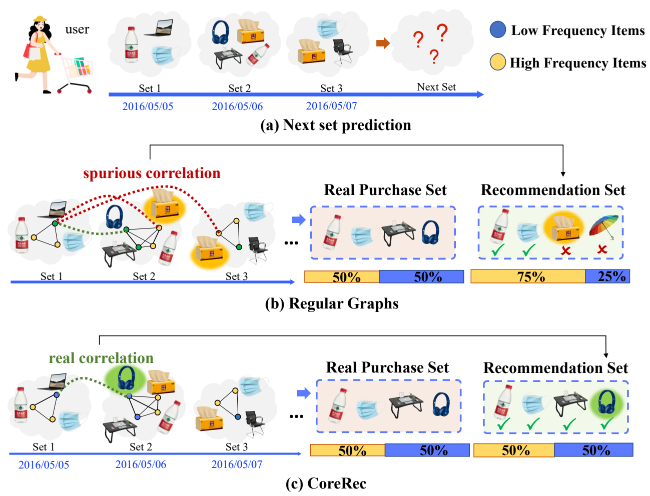


Figure 1: Our objective is to forecast succeeding sets from a given sequence of sets. Our investigation reveals that leveraging homogeneous information within the sets to establish connections inadvertently inflates the occurrence of high-frequency objects. To address this, we propose the incorporation of heterogeneous information and causal graphs, which effectively rectify the bias stemming from the amplified prevalence of high-frequency objects.

Prior research (Hu and He 2019; Jung et al. 2021; Li et al. 2023; Qin, Wang, and Li 2021; Sun et al. 2020; Wang et al. 2020; Yu et al. 2022a, 2023, 2022b) has predominantly focused on investigating intra-set and inter-set correlations among items. This endeavor aims to enhance the representation of items, facilitating a more nuanced understanding of user preferences. Notably, Yu et al. introduced temporal graphs to establish connections among items within the same set, thereby modeling intra-set relationships (Yu et al. 2020). In a similar vein, Sun et al. utilized a co-transformer framework to aggregate item representations at both intra-set and inter-set levels (Sun et al. 2020).

In spite of the progress achieved in existing studies, there exist two pivotal concerns that impede the precise model-

[*]The corresponding author.

ing of item correlations. Firstly, inter-set correlations, lacking inherent labels, remain implicit. This could lead models to erroneously connect unrelated items, possibly assimilating spurious correlations within item representation learning. Secondly, among individual users, numerous high-frequency items recurrently appear across historical sets. For instance, as illustrated in Figure 1(a), items such as "napkin" and "gauze mask" are pervasive but not necessarily correlated, unlike less frequent items like "laptop" and "headset". Notably, the dominance of high-frequency items due to their extensive co-occurrence might distort correlation modeling, particularly affecting low-frequency items. Consequently, the inadvertent absorption of misleading correlations during item representation learning can lead to susceptibility in the accuracy of recommending the next set—potentially suppressing low-frequency items in such recommendations. Given the continuous real-time refinement of real-world recommendations through user interactions, this phenomenon can trigger the Matthew effect and consequent performance degradation (Wang et al. 2021a).

Our objective is to mitigate spurious inter-set correlations. A pivotal aspect of this objective involves scrutinizing the impact of potentially correlated items on both item-level representation learning and prediction. This scrutiny inherently entails estimating the causal effects of correlations on the prediction of subsequent sets. To accomplish this, we adopt the causal graph framework (Feng et al. 2021; Jr. 2005; Wang et al. 2021b; Wei et al. 2021; Zhang et al. 2021a,b) to delineate the causal relationships within the context of the next set recommendation (depicted in Figure 4).

The crux of our approach involves creating a counterfactual scenario for each item after an intervention. In this scenario, inter-set correlations are effectively obstructed, compelling the model to depend solely on intra-set correlations for item representation learning. Through this, we can ascertain the causal effect of inter-set correlations by comparing the predictions made under standard conditions with those made in the post-intervention situation.

Technically, we propose a **Co**unterfactual **Co**rrelation Infe**re**nce framework, namely **CoreRec**, which operates at the intersection of counterfactual analysis and correlation inference. CoreRec is designed to dissect and harness inter-set correlations in a controlled manner, enabling us to illuminate causal relationships within complex systems. The framework employs two distinct weighted graphs: a regular graph capturing regular inter-set correlations and an intervened graph with these correlations strategically suppressed (depicted in Figure 1). Furthermore, CoreRec invokes a causal intervention that perturbs the aggregation mechanism, compelling the model to pivot towards utilizing the inherent intra-set correlation attributes of individual items. To operationalize this, we introduce a switch module that judiciously toggles between the intervened and regular item representations, taking into account multifaceted determinants such as causal influence and prediction reliability.

Extensive experiments are conducted to validate the effectiveness of our framework. Our CoreRec achieves state-of-the-art performance on three commonly used datasets. In particular, CoreRec achieves 61.46 PHR@20 and 47.28 Recall@20 on the JD dataset (*vs*. ETGNN: 56.52 PHR@20 & 38.12 Recall@20), which increased by 8.74% and 22.77% respectively compared with the ETGNN.

In summary, our contributions are three-fold as follows:

- We identify two critical issues that hinder accurate inter-set correlation modeling, and formulate the causal graph of correlation-based next set recommendation.
- We propose a novel framework named CoreRec, which constructs two weighted graphs and a switch module, together formulating counterfactual inference and achieving adaptive inter-set correlation modeling.
- We conduct extensive experiments on three real-world datasets and the experimental results strongly demonstrate that CoreRec outperforms all the baseline methods.

Related Work

Next Set Prediction is increasingly receiving attention in recommendation system research (Hu et al. 2020; Li et al. 2023; Qin, Wang, and Li 2021; Yu et al. 2023). Especially, Rendle *et al.* (Rendle, Freudenthaler, and Schmidt-Thieme 2010) proposed a classical method to recommend the next basket. It learns both sequential behaviors and personal tastes, based on personalized transition graphs over underlying Markov chains. Recently, Yu *et al.* (Yu et al. 2020) adopted a method that learns element relationships based on a set-level co-occurrence graph and uses attention-based temporal dependency learning for the next set prediction. However, previous methods lack a meticulous examination of the authenticity of these presumed associations. To address this gap, we introduce a discerning switch model imbued with counterfactual techniques, which critically evaluates the essentiality of these associations.

Causality-aware Model Prediction. Causal inference finds extensive utility across a spectrum of machine learning domains. In the context of recommendation, the realm of causal inference (Pearl 2009) predominantly centers on mitigating diverse biases intrinsic to user feedback. This encompasses addressing position bias (Joachims, Swaminathan, and Schnabel 2017; Shengyu et al. 2023; Wang et al. 2021c; Zhang et al. 2021c), countering clickbait-related concerns, and alleviating popularity-induced bias. For example, Saito et al. calculated exposure propensity for individual user-item pairs and employed sample re-weighting to tackle the challenge of non-random missing data (Saito et al. 2020). Nevertheless, the existing methods heavily lean on the precision of propensity estimation, often grappled by elevated propensity variance. Consequently, we focus on the switch techniques that are frequently enlisted as a subsequent remedy.

Problem Formalization

Let $U = \left\{u_1, \cdots, u_{|U|}\right\}$ denote a collection of users, $V = \left\{v_1, \cdots, v_{|V|}\right\}$ denote all the available items, and $\mathbf{E} \in \mathbb{R}^{|V| \times d}$ denote the embedding matrix of all items. We use $\hat{S}_i = \left[v_i^1, v_i^2, \cdots, v_{|\hat{S}_i|}^t\right]$ to denote a user's historical interaction and organize temporal sets $S_i = \left[s_i^1, s_i^2, \cdots, s_i^t\right]$ by treating items bought in the same timestamp as a set, in which $s_i^t = \left\{v_1, v_2, \cdots v_{|s_i^t|}\right\}$ represents a set interacted by the user u_i at time t. For a user, given the user's historical interaction temporal sets S_i, the goal is to predict the next set according to the historical records, that is,

$$s_i^{t+1} = F\left(S_i, \mathbf{W}\right), \tag{1}$$

where $\mathbf{W}$ represents the trainable parameters.

Approach

Previous methods often employed fixed item-associations to formulate predictions. However, this rigid approach, characterized by predetermined artificial associations, hinders adaptability to data-specific nuances. CoreRec introduces a dynamic solution by training the switch model to grasp distinct item associations inherent in each dataset, thus yielding data-adaptive item representations. CoreRec comprises Graph-based Item Representation Learning and Switch Designed by Counterfactual Intervention. The method begins by constructing two graphs to delineate standard and post-intervention conditions. Next, a switch model is trained through causal reasoning on historical sequences. This culminates in the prediction of forthcoming sets, anchored in the item representation selected by the switch mechanism.

Graph-based Item Representation Learning

We designed two weight graphs: the regular graph encompassing connections among items intra and inter-set, and the intervened graph encompassing solely the intra-set connections. During the item representation learning process, we comprehensively incorporate factors like purchase time intervals, long- and short-term sequences.

Time Interval Matrices for Graph Construction. To reinforce the user's recently active interest representation, we extract a short-term sequence from the long-term original sequence and then obtain two input sequences. We use the $S_i^L = [s_i^1 \dots s_i^{t/2} \dots s_i^t]$ to denote the long-term sequence, and use the $S_i^S = [s_i^{t/2} \dots s_i^t]$ to denote the extracted short-term sequence. Our investigation reveals that the sets with extended time intervals display a comparatively diminished correlation. Capitalizing on this discernment, we formulate a time interval matrix to encapsulate the influence of temporal gaps between items. Note that the distance between two items in the same set is 0, and the distance between items in set s_i^j and set s_i^k can be denoted as $|j-k|$.

For arbitrary nodes $v_n^j \in s_i^j$ and $v_m^k \in s_i^k$, the time interval value is defined as $\delta_{n,m}$, which is negatively correlated with the distance, *i.e.*,

$$\delta_{n,m} = |S_i^*| - |j-k|, \tag{2}$$

where $S_i^* \in \{S_i^L, S_i^S\}$ denotes long-term or short-term sequence of user i, and $|S_i^*|$ denotes the total number of sets in current sequence. Therefore, the time-interval matrix is naturally constructed according to $\{\delta_{n,m}\}$. Specially, we consider the time-interval values for the set-self and all the 2-set pairs, which results in $C^2_{|S_i^*|} + 1$ time-interval matrices $\{\mathbf{T}_{i,k}\}$, where C_n^k represents the number of k-combinations from a given set of n elements. Finally, we average all of these time-interval matrices and normalize it w.r.t its maximum value to obtain the time-interval matrix $\mathbf{T_i}$,

Regular Graphs Construction. We take the sequence that contains three sets as an example, as is shown in Figure 3. First, we define the purchase frequency of item v_i as:

$$f_{v_i} = D_{v_i} / \sum_{v_j \in \hat{S}_i} D_{v_j}, \tag{3}$$

where D_{v_i} denotes the number of interactions in sequence for item v_i. Then, we divide the items into high-frequency items and low-frequency items with ε as a threshold:

$$v_{i,*} = \begin{cases} v_{i,*}^{high} & \text{if } f_{v_i} > \varepsilon, \\ v_{i,*}^{low} & \text{otherwise.} \end{cases} \tag{4}$$

We connect the items in the same set and also connect all the low-frequency items $v_{i,*}^{low}$ between any two different sets. We count the number of co-occurrence pairs to obtain the co-occurrence matrix $\mathbf{C}_i^{regular}$ for the regular graph. Then, the weighted matrix for the regular graph can be obtained,

$$\mathbf{W}_i^{regular} = \mathrm{Norm}_{\max}\left(\mathbf{C}_i^{regular}\right) + \lambda \mathbf{T_i} \odot (\mathbf{C}_i^{regular} > 0), \tag{5}$$

where $\odot$ indicates element-wise product, $\mathbf{C}_i^{regular}$ denotes the co-occurrence matrix of the intervened graph, and $\mathbf{T_i}$ denotes the time interval matrix. The hyper-parameter λ can control the contribution of the time interval matrix. Finally, we add self-connection to the weighted matrix and construct a regular graph concerning the weighted matrix.

Intervened Graphs Construction. We connect all the items in the same set and obtain three fully connected graphs. Then we combine the same items from different graphs and obtain the co-occurrence matrix $\mathbf{C}_i^{inter}$. The weighted matrix $\mathbf{W}_i^{inter}$ is obtained by integrating the co-occurrence matrix $\mathbf{C}_i^{inter}$ and the time-interval matrix $\mathbf{T_i}$,

$$\mathbf{W}_i^{inter} = \mathrm{Norm}_{\max}\left(\mathbf{C}_i^{inter}\right) + \lambda \mathbf{T_i} \odot (\mathbf{C}_i^{inter} > 0), \tag{6}$$

where $\odot$ indicates element-wise product, and the hyper-parameter λ control the scale of the time interval matrix. We further analyze the variation of hyper-parameter values λ in the experimental part. We also add self-connection for each item appearing in sequence, which helps to reduce information loss. In this way, we construct an intervened graph with weights from the weighted matrix.

GNNs for Feature Encoding. We perform GNNs on the above two types of weighted graphs. Let $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ denotes the graph with a weighted matrix $\mathbf{W} \in \mathbb{R}^{|\mathcal{V}_i| \times |\mathcal{V}_i|}$, where $\mathcal{V}_i$ denotes the items in S_i and $\mathcal{E}_i$ denotes the edges in $\mathcal{G}_i$. Each item in graph $\mathcal{G}_i$ is linearly combined according to the attention score. Let $\mathcal{N}(m)$ be the set of neighborhood nodes of v_m, and $\mathbf{h}_{\mathcal{N}(m)}$ denotes the neighborhood representation, *i.e.*,

$$\mathbf{h}_{\mathcal{N}(m)} = \sum_{v_n \in \mathcal{N}(m)} \pi\left(v_m, v_n\right) \mathbf{h}_{v_n}, \tag{7}$$

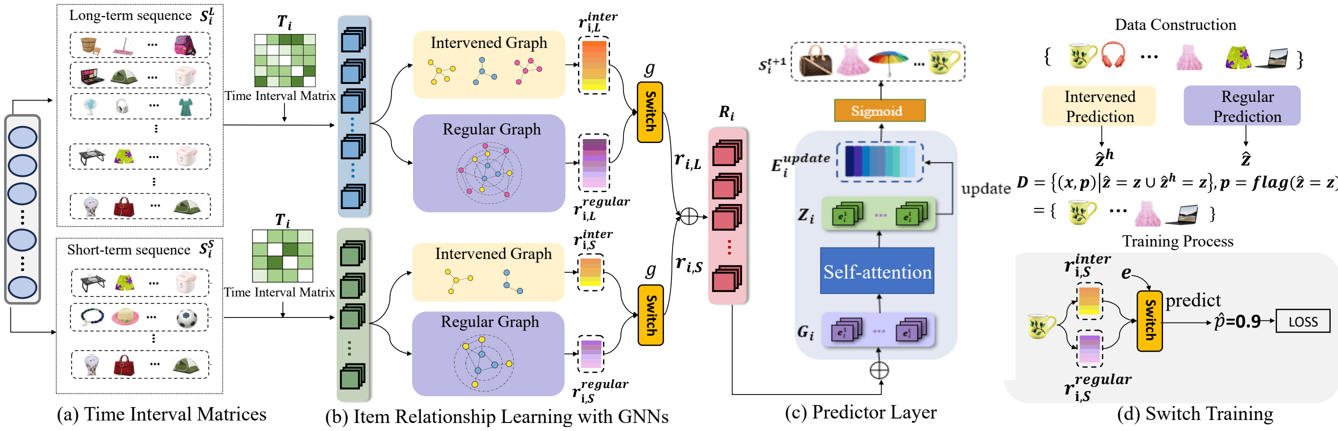


Figure 2: Overview of CoreRec. (a) Construct time interval matrices to capture temporal dependency. (b) Construct regular graphs and intervened graphs to obtain the sequence of item representation $r_{i,*}^{regular}$ and $r_{i,*}^{inter}$ respectively. (c) Choose the final sequence of item representations R_i by the switch model. (d) Train the switch by applying causal intervention.

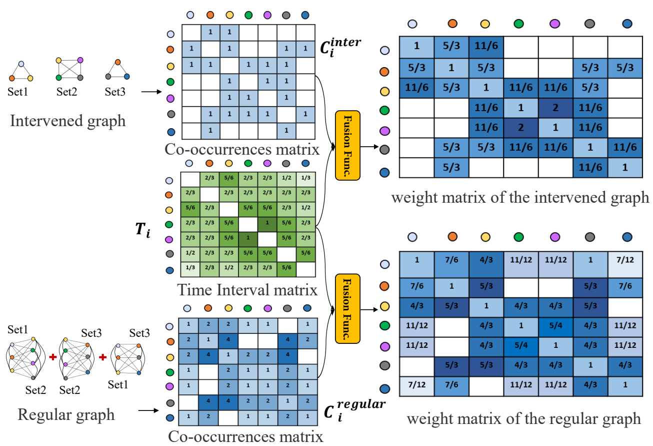


Figure 3: Intervened graph and regular graph construction. The 'Fusion Func.' is stated as Eq. (6) and Eq. (5)

.

where $\pi(v_m, v_n)$ estimates the importance weight of different neighbors. We implement $\pi(v_m, v_n)$ as follows,

$$\pi(v_m, v_n) = Relu\left(\mathbf{w}_1\left[\left(\mathbf{h}_{v_m} \odot \mathbf{h}_{v_n}\right) \| \hat{w}_{mn}\right]\right), \quad (8)$$

where $\|$ indicates concatenation operation, $\hat{w}_{mn}$ denotes the weight of edges in graph, $\mathbf{w}_1$ denotes trainable parameters.

We concat the item representation $\mathbf{h}_{v_m}$ and its neighborhood representation $\mathbf{h}_{\mathcal{N}(m)}$ to obtain the final sequence of item's representation $\mathbf{R}_i = \{\mathbf{h}_{v_m} \mid m = 1, \ldots, |\hat{S}_i|\}$

Next Set Prediction. From the above two weighted graphs, we can obtain two sequences of the item's representation, $r_{i,*}^{regular}$ and $r_{i,*}^{inter}$, we then employ the post-trained switch to choose one sequence of item's representation $\mathbf{R}_i$. Existing methods usually utilize set embedding by pooling operation, but it will cause information loss, so we directly use the sequence of the item's representation $\mathbf{R}_i$ as input. We perform the self-attention to capture temporal dependency,

$$\mathbf{Z}_i = softmax\left(\frac{(\mathbf{R}_i W_q)\cdot(\mathbf{R}_i W_k)^\top}{\sqrt{d_k}}\right)\cdot(\mathbf{R}_i W_v), \quad (9)$$

where W_q,W_k and W_v are trainable parameters. Then we empoly $\mathbf{Z}_i$ to update the item original embedding matrix $\mathbf{E}_i$,

$$\mathbf{E}_{i,I(j)}^{\text{update}} = \mathbf{E}_{i,I(j)} + \mathbf{Z}_{i,j}, \quad (10)$$

where I(·) is a function that maps items $v_{i,j}$ to its corresponding index in $\mathbf{E}_i$. In Equation (10), the item representations are updated according to both the co-occurrence relationships and the temporal dependency of the items. We maintain the original representations for all other items. The probabilities of each item appearing in the subsequent set can be computed based on the current state,

$$\hat{y}_i = sigmoid\left(\mathbf{E}_i^{\text{update}} \cdot \boldsymbol{w}_o + \boldsymbol{b}_o\right), \quad (11)$$

where $\boldsymbol{w}_o \in \mathbb{R}^d$ and $b_o \in \mathbb{R}$ are trainable parameters to provide the final prediction result. When training, predicting the next set could be treated as a multi-label learning problem (Ghamrawi and McCallum; Yu et al.; Zhang and Zhou; Zhang and Zhou), so we adopt the loss function with L2 regularization technique as follows.

$$L = -\frac{1}{N}\sum_{i}^{N}\frac{1}{|V|}\sum_{j}^{|V|} y_i^j \log\left(\hat{y}_i^j\right) + \left(1 - y_i^j\right)\log\left(1 - \hat{y}_i^j\right) + \gamma\|W\|^2, \quad (12)$$

where N denotes the number of training samples, y_i and $\hat{y}_i$ denote the ground truth and the predicted appearing possibility in the next set of user u_i, and γ is a hyper-parameter to control the importance of L_2 regularization.

Switch Designed by Counterfactual Intervention

In recommendation systems, the item correlation across diverse datasets varies significantly, shaped by the unique attributes of the generating users. However, previous approaches have been restricted to employing static patterns

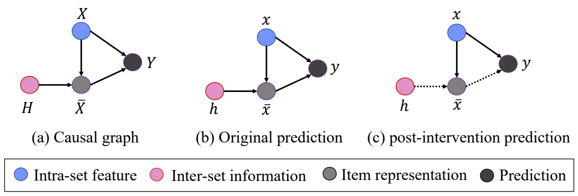


Figure 4: Cause-effect view of set recommendation.

for designing and establishing associations between items. These methods struggle to flexibly grasp the distinct data structures. To address this limitation, we advocate for an approach where the choice of item representation is informed by a switch model trained on historical datasets.

Causal Graph. The underpinning causal relationships among variables are elegantly captured through a causal graph, as illustrated in Figure 4. In the context of our study, we conceptualize the inference of the next set prediction framework within the framework of a causal graph. This representation encapsulates four key variables:

- X, which denotes the item representation from the graph based on the intra-set correlations.
- H, which denotes the inter-set information.
- $\bar{X}$, which denotes final item representation.
- Y, which denotes the prediction result.

The final item representation $\bar{X}$ is directly affected by the intra-set item representation X and the inter-set information H. Existing set prediction methods can be divided into two categories. One approach simply aggregates item embeddings into set embeddings for prediction, directly utilizing the information within the set, in which, the intra-set information X between items directly impacts the results Y. The other approach uses a sequence of item embeddings for prediction, considering both intra- and inter-set information. As the inter-set information between items is introduced only when updating item embeddings for prediction, there is no direct effect between H and Y. Moreover, the temporal sets prediction result Y is directly affected by the final item representation $\bar{X}$, which is represented as $\bar{X} \longrightarrow Y$.

Causal Intervention. We utilize causal intervention (Galles and Pearl 2013; Mueller, Li, and Pearl 2021; Pearl 2012) to assess the causal effect of inter-set information on the prediction (i.e., the causal effect of $H = h$). This method entails assigning an instance to a treatment variable deliberately. In our causal graph described earlier, the causal effect denoted as **e** is precisely defined as:

$$\begin{aligned} \boldsymbol{e} &= f(\boldsymbol{x}, h \mid \hat{\theta}) - f(\boldsymbol{x}, \mathrm{do}(H = \emptyset) \mid \hat{\theta}) \\ &= f(\boldsymbol{x}, h \mid \hat{\theta}) - f(\boldsymbol{x}, \emptyset \mid \hat{\theta}) \\ &= \hat{\boldsymbol{y}} - \hat{\boldsymbol{y}}^{\mathrm{h}}, \end{aligned} \tag{13}$$

where the term $\mathrm{do}(H = \emptyset)$ denotes a forceful causal intervention, assigning a reference status of H. This intervention yields a post-intervention prediction $f(\boldsymbol{x}, h \mid \hat{\theta})$ (see Figure 4(c)). As H lacks a predecessor, $f(\boldsymbol{x}, \mathrm{do}(H = \emptyset) \mid \hat{\theta})\& = f(\boldsymbol{x}, \emptyset \mid \hat{\theta})$, expressed as $\hat{\boldsymbol{y}}^{\mathrm{h}}$. Intuitively, this post-intervention prediction signifies the outcome if the inter-set information is absent from the target item representation. We contend that $\boldsymbol{e}$ offers insights into selecting a more expressive item representation for the target item.

Algorithm 1: CoreRec Training

Input: Training data set sequence S_{train}
Output: $\hat{\theta}$, the parameter for $GNN(\cdot)$; $\hat{\boldsymbol{\eta}}$, the parameter for $g(\cdot)$

Optimize Eq.(11) with S_{train}, obtain $\hat{\theta}$; ▷ *Training GNN*
Construct data $\mathcal{D}$, including regular and intervened cases
Calculate causal effect $\boldsymbol{e}$; ▷ *Causal Intervention*
Optimize Eq.(17), obtain $\hat{\boldsymbol{\eta}}$; ▷ *Training Switch Model*
Return $\hat{\theta}$ and $\hat{\boldsymbol{\eta}}$;

Algorithm 2: CoreRec Inference

Input: Testing data set sequence S_{test}, parameter $\hat{\theta}$ and $\hat{\boldsymbol{\eta}}$
Output: Binary prediction results for each item y

Extract feature through GNN($\hat{\theta}$) with S_{test}; ▷ *Encoding*
Calculate $f(\boldsymbol{x}, h \mid \hat{\theta})$; ▷ *Regular prediction*
Calculate $f(\boldsymbol{x}, \emptyset \mid \hat{\theta})$; ▷ *Post-intervention prediction*
Calculate causal effect $\boldsymbol{e}$;
Calculate final item embedding through Eq.(15) with $\hat{\boldsymbol{\eta}}$;
Return final classification y ▷ *Final prediction*

Switch Model. We train the set prediction model with the regular graph and the intervened graph respectively. To construct the training data, we calculate the ground truth according to the correctness of $\hat{z}$ and $\hat{z}^{\mathrm{h}}$, where $\hat{z} = \arg\max \hat{\boldsymbol{y}}$ and $\hat{z}^h = \arg\max \hat{\boldsymbol{y}}^h$, see Figure 2 (d).

Then we train the switch model by fixing the parameters of GNNs with the two item representations $r_{i,*}^{regular}$ and $r_{i,*}^{inter}$, and the causal effect $\boldsymbol{e}$. We devise the switch model as a multi-layer perceptron to make wise choices between $r_{i,*}^{regular}$ and $r_{i,*}^{inter}$,

$$r_{i,*} = \begin{cases} r_{i,*}^{regular}, \hat{p} \geq \kappa, \\ r_{i,*}^{inter}, \hat{p} < \kappa, \end{cases} \quad \hat{p} = g\left(r_{i,*}^{regular}, r_{i,*}^{inter}, \boldsymbol{e} \mid \boldsymbol{\eta}\right), \tag{14}$$

where $g(\cdot)$ represents a binary classifier parameterized by $\boldsymbol{\eta}$. The classifier's output, denoted as $\hat{p}$, guides the decision-making process, with κ serving as the decision threshold.

The training of the switch model is thus formulated as:

$$\hat{\boldsymbol{\eta}} = \min_{\boldsymbol{\eta}} \sum_{(\boldsymbol{x}, p) \in \mathcal{D}} l(\hat{p}, p), \tag{15}$$

where p and $\hat{p}$ denote the ground truth and the predicted appearing possibility.

EXPERIMENTS

To evaluate the effectiveness of the proposed method, we conduct experiments on three real-world datasets:

TaFeng (TF)[1]: The TaFeng dataset is a public dataset that contains a Chinese grocery store transaction data from

[1] www.kaggle.com/chiranjivdas09/ta-feng-grocery-dataset

	Method	K=20			K=40			K=60			K=80		
		PHR	NDCG	Recall	PHR	NDCG	Recall	PHR	NDCG	Recall	PHR	NDCG	Recall
TaFeng	TOP	35.45	9.21	10.17	44.65	10.24	13.09	52.77	11.32	16.77	57.82	11.98	19.03
	DeepFM	46.18	12.37	16.68	57.48	14.04	22.57	62.38	14.45	23.82	65.98	14.73	24.46
	Sets2Sets	49.86	13.56	17.79	61.01	14.94	23.44	70.45	15.78	25.15	72.02	16.65	27.33
	DSNTSP	60.01	16.40	21.50	68.69	18.56	26.57	70.90	19.42	28.54	72.05	19.57	29.65
	DNNTSP	**62.77**	16.45	22.13	69.70	18.09	26.96	72.67	19.43	29.37	74.25	19.44	31.14
	ETGNN	62.45	16.47	21.06	69.73	18.67	27.13	71.88	19.52	29.65	73.92	19.53	31.29
	CoreRec	62.38	**16.91**	**22.27**	**70.39**	**18.73**	**27.45**	**74.46**	**19.70**	**30.59**	**76.34**	**20.22**	**32.30**
	improv.(%)	-0.63	**2.67**	**0.63**	**0.95**	**0.32**	**1.18**	**2.46**	**0.92**	**3.17**	**2.81**	**3.32**	**3.23**
TaoBao	TOP	3.99	0.35	0.50	5.03	0.39	0.71	6.71	0.46	0.98	8.39	0.51	1.11
	DeepFM	22.09	2.98	2.71	27.61	3.27	3.71	29.90	3.37	4.34	31.57	3.44	4.71
	Sets2Sets	23.95	3.62	4.83	31.29	4.21	6.62	34.43	4.56	7.75	35.63	4.77	8.41
	DSNTSP	29.76	4.66	6.04	39.71	5.06	7.69	45.73	5.25	8.61	48.83	5.40	9.33
	DNNTSP	29.69	4.70	5.83	41.50	5.39	8.29	47.16	5.88	9.95	49.68	6.16	10.97
	ETGNN	32.52	5.60	6.24	42.87	5.78	8.45	47.93	6.12	10.23	50.01	6.42	11.32
	CoreRec	**36.05**	**6.07**	**7.63**	**46.33**	**6.73**	**10.48**	**49.68**	**7.11**	**11.63**	**50.94**	**7.34**	**12.30**
	improv.(%)	**10.85**	**8.39**	**26.3**	**8.07**	**16.44**	**26.4**	**3.65**	**16.18**	**16.8**	**1.86**	**14.33**	**12.1**
JingDong	TOP	21.87	6.48	14.19	31.77	8.17	21.46	35.42	8.76	24.21	40.10	9.43	27.73
	DeepFM	40.57	15.02	23.85	49.34	16.40	27.84	55.34	17.50	34.47	56.26	17.85	35.91
	Sets2Sets	48.96	17.71	34.12	54.69	21.37	40.28	59.03	22.97	46.05	60.33	23.58	48.04
	DSNTSP	50.52	24.36	38.51	64.45	25.09	46.59	69.21	25.54	48.84	71.72	26.39	55.15
	DNNTSP	53.12	22.06	36.36	65.10	24.52	46.00	67.70	25.71	50.08	71.87	26.87	56.45
	ETGNN	56.52	24.26	38.12	67.11	25.78	47.97	69.83	27.11	52.38	73.43	27.43	57.83
	CoreRec	**61.46**	**28.37**	**47.28**	**70.31**	**30.51**	**55.36**	**76.04**	**31.82**	**61.11**	**78.13**	**32.45**	**64.05**
	improv.(%)	**8.74**	**16.46**	**22.77**	**4.77**	**18.35**	**15.41**	**8.89**	**17.37**	**16.67**	**6.41**	**18.30**	**10.76**

Table 1: Comparisons with methods on Top-K performance. Note that the bold values indicate the best score, and the underlined value means the best among the baselines. The improvement (%) of our CoreRec is based on the score with an underline.

November 2000 to February 2001. We remove users whose purchase time is less than 10 days.

TaoBao (TB)[2]: This dataset is a subset of Taobao user behavior data (Zhu et al.) that contains behaviors including click, purchase, adding item to the shopping cart and item favoring. We select all purchasing behaviors.

Jingdong (JD)[3]: The JingDong dataset contains user action records from February 1, 2018, to April 15, 2018. We remove users whose purchase time is less than 5 days.

For the readers' convenience, we provide the statistics of the three datasets in Table 2.

Data	items	sets	users	cate
TF	21,858	76,251	8,816	1,954
TB	242,111	34,642	4,827	5,070
JD	41,212	270,397	2,011	6

Table 2: Statistic information after pre-processing on three datasets. Note that "I/S" indicates the average ratio of items to sets, and "S/U" is the average ratio of sets to users.

Implementation Details. We treat all the items bought in the same order as a set and divide each dataset into train, validation, and test sets across users with ratios of 80%, 10%, and 10%. For evaluation, we generate a ranking list of top-K items from the output and K is from 10 to 100, with an interval of 10. The epoch is set to 100, 300, and 500 on Tafeng, Taobao, and JD datasets respectively. We adopt Adam (Kingma and Ba) with a learning rate set to 0.001 as the optimizer in the experiment. In addition, the hidden dimension and batch size are set to 32 and 64. Furthermore, λ in Eq. (6) and in Eq. (5) are set to 0.4, and γ in Eq. (12) is set to 0.8. Besides, we set the short sequence whose length is 1/2 of the original sequence. We use PyTorch (Paszke et al.) to implement our model and train it on 4 GeForce GTX 1080Ti GPUs. Regarding the metrics, we leverage Personal Hit Ratio@K (PHR@K), Normalized Discounted Cumulative Gain@K (NDCG@K), and Recall@K, to evaluate the performance of temporal set prediction.

[2]tianchi.aliyun.com/dataset/dataDetail?dataId=649

[3]jdata.jd.com/html/detail.html?id=8

Comparison

To demonstrate the effectiveness of CoreRec, we compare it with six competing baselines, including DeepFM (Guo et al.), Sets2Sets (Hu and He), DSNTSP (Sun et al.), DNNTSP (Yu et al.), and ETGNN (Yu et al.). To ensure fair comparison, we use the same setting for all methods and train all models from scratch. The experimental results are summarized in Table 1. ETGNN demonstrates a significantly enhanced performance in contrast to TOP, DeepFM, Sets2Sets, DSNTSP, and DNNTSP. Nonetheless, our novel CoreRec outperforms the ETGNN. Notably, the JingDong and Taobao datasets encompass a broader spectrum of items, characterized by a pronounced disparity be-

TaFeng	K=50			K=100		
	PHR	NDCG	Recall	PHR	NDCG	Recall
w/o s	72.67	18.78	28.75	77.33	20.21	33.43
w/o t	72.08	18.57	28.92	77.23	20.01	33.71
w/o G_{reg}	71.78	18.59	28.45	75.94	19.96	32.94
w/o G_{inter}	72.18	18.54	28.85	76.53	19.91	33.39
w/o switch	72.06	18.67	28.59	76.43	20.13	33.12
CoreRec	**73.07**	**19.26**	**29.15**	**77.33**	**20.74**	**34.06**
TaoBao	K=50			K=100		
	PHR	NDCG	Recall	PHR	NDCG	Recall
w/o s	44.44	6.81	9.91	51.36	7.46	11.95
w/o t	46.54	6.19	10.33	52.62	6.87	12.56
w/o G_{reg}	45.49	5.68	9.27	51.32	6.58	11.97
w/o G_{inter}	47.37	6.81	10.69	51.24	7.45	12.65
w/o switch	45.58	6.25	9.32	51.41	6.92	12.03
CoreRec	**47.58**	**6.90**	**11.04**	**52.62**	**7.47**	**12.67**
JingDong	K=50			K=100		
	PHR	NDCG	Recall	PHR	NDCG	Recall
w/o s	68.22	25.52	52.70	77.08	27.94	64.26
w/o t	70.31	25.39	55.64	77.08	30.84	64.18
w/o G_{reg}	66.14	25.16	48.46	75.12	27.79	62.45
w/o G_{inter}	70.31	25.28	53.58	**80.21**	27.42	64.34
w/o switch	69.22	28.81	55.59	76.02	30.15	64.02
CoreRec	**73.96**	**31.37**	**59.20**	79.69	**32.86**	**66.21**

Table 3: Ablation study of CoreRec. s denotes short-term sequence, t denotes time interval matrices, G_{reg} denotes the regular graph, G_{inter} denotes the intervened graph.

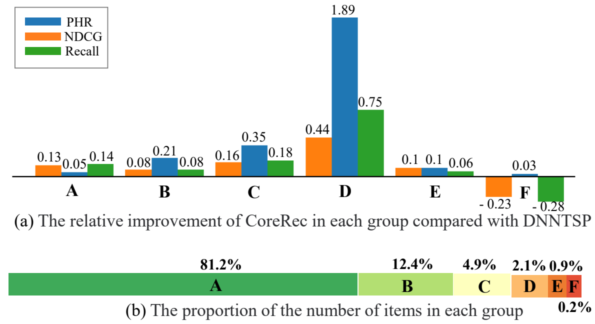
PHR@10	A	B	C	D	E	F
DNNTSP	5.07	10.41	12.77	18.24	25.09	54.55
CoreRec	**5.12**	**10.62**	**13.12**	**20.13**	**25.19**	**54.58**
NDCG@10	A	B	C	D	E	F
DNNTSP	2.28	3.97	3.94	6.84	9.01	**29.12**
CoreRec	**2.41**	**4.05**	**4.10**	**7.28**	**9.11**	28.89
Recall@10	A	B	C	D	E	F
DNNTSP	2.89	7.16	7.58	11.33	16.54	**45.55**
CoreRec	**3.03**	**7.24**	**7.76**	**12.08**	**16.60**	45.27

Table 4: The performance varies on different purchase frequency groups on Tafeng.

tween low-frequency and high-frequency items. This distinction accentuates the susceptibility to spurious correlations. Consequently, our model excels on the JingDong and Taobao datasets by mitigating such spurious correlations through the utilization of the switch module, as illustrated in Table 1. Furthermore, the data presented in Table 3 underscores the more discernible impact of incorporating time interval matrices on the JingDong dataset.

Ablation Study

To investigate the efficacy of key modules in CoreRec, we consider several model variants and obtain the conclusions: *1) Impact of the Short Sequence.* Table 3 shows that short-term sequences help to achieve better performance. Because the short-term sequence represents the user's recent active interests. *2) Impact of the Time Interval Matrices.* We consider that two sets with small time intervals have a strong correlation, while two sets with large time intervals have a weak correlation. Table 3 shows that time interval matrices are effective. *3) Impact of the Regular Graph.* The regular graph not only explores the items' intra-set correlation but also considers the items' purchase frequency and items' inter-set correlation, which are effective on the whole. *4) Impact of the Intervened Graph.* The intervened graph connects the items in the same set, which formulates a counterfactual situation. From Table 3, we can learn that items' intra-set correlations are also effective on the whole. *5) Impact of the Post-trained Switch Module.* Switch module aims is to determine whether to aggregate information across sets. Compared with the undifferentiated aggregation method, using the switch can dynamically select the more appropriate representation for each item.



(a) The relative improvement of CoreRec in each group compared with DNNTSP

(b) The proportion of the number of items in each group

Figure 5: We divide items into groups based on items' purchased frequency. 'A'(0-20) group contains items purchased 0-20 times, and for easy to write, 'B'(20-50), 'C'(50-100), 'D'(100-200), 'E'(200-500), 'F'(500+).

Performance Vary on Purchase Frequency. Initial analysis of the data reveals a prevalent long-tail distribution in item frequency across multiple scenarios. This distribution raises concerns about low-frequency suppression, wherein items appearing in a limited number of sets establish fewer connections, resulting in diminished recommendations. To address this challenge, our CoreRec offers a viable solution. Insights drawn from our analysis, as depicted in Figure X and Table Y, highlight that items with fewer than 500 purchases (encompassing Groups A, B, C, D, and E) constitute a substantial majority (99.8%). In this context, our proposed CoreRec surpasses DNNTSP in performance for these items. Conversely, for items exceeding 500 purchases, CoreRec performs comparably to DNNTSP.

Conclusion

In this study, we delve into the detrimental impact of spurious inter-set correlations on model performance. These correlations often emerge from the confluence of high-frequency items and extraneous noise. To address this challenge, we introduce CoreRec, an innovative framework designed to mitigate these spurious inter-set correlations. CoreRec draws upon the principles of causal intervention, manifested through two distinct graphs: an intervened graph and a regular graph. By incorporating a purpose-built switch, CoreRec adeptly navigates between post-intervention predictions and original predictions. Extensive experimentation across three benchmark datasets robustly demonstrates the effectiveness of our proposed CoreRec framework.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 62337001, 62376243, 62037001), Young Elite Scientists Sponsorship Program by CAST (2021QNRC001)

References

Feng, F.; Huang, W.; He, X.; Xin, X.; Wang, Q.; and Chua, T. 2021. Should Graph Convolution Trust Neighbors? A Simple Causal Inference Method. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 1208–1218. ACM.

Galles, D.; and Pearl, J. 2013. Testing Identifiability of Causal Effects. *CoRR*, abs/1302.4948.

Ghamrawi, N.; and McCallum, A. 2005. Collective multi-label classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, 195–200.

Guo, H.; Tang, R.; Ye, Y.; Li, Z.; and He, X. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In Sierra, C., ed., *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, 1725–1731. ijcai.org.

Hu, H.; and He, X. 2019. Sets2sets: Learning from sequential sets with neural networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1491–1499.

Hu, H.; He, X.; Gao, J.; and Zhang, Z. 2020. Modeling Personalized Item Frequency Information for Next-basket Recommendation. In Huang, J.; Chang, Y.; Cheng, X.; Kamps, J.; Murdock, V.; Wen, J.; and Liu, Y., eds., *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, 1071–1080. ACM.

Joachims, T.; Swaminathan, A.; and Schnabel, T. 2017. Unbiased learning-to-rank with biased feedback. In *Proceedings of the tenth ACM international conference on web search and data mining*, 781–789.

Jr., H. E. K. 2005. Judea Pearl, *Causality*, Cambridge University Press (2000). *Artif. Intell.*, 169(2): 174–179.

Jung, S.; Park, Y.; Jeong, J.; Kim, K.; Kim, H.; Kim, M.; and Kwak, H. 2021. Global-Local Item Embedding for Temporal Set Prediction. In Pampín, H. J. C.; Larson, M. A.; Willemsen, M. C.; Konstan, J. A.; McAuley, J. J.; Garcia-Gathright, J.; Huurnink, B.; and Oldridge, E., eds., *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*, 674–679. ACM.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, M.; Jullien, S.; Ariannezhad, M.; and de Rijke, M. 2023. A next basket recommendation reality check. *ACM Transactions on Information Systems*, 41(4): 1–29.

Mueller, S.; Li, A.; and Pearl, J. 2021. Causes of Effects: Learning individual responses from population data.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32: 8026–8037.

Pearl, J. 2009. *Causality*. Cambridge university press.

Pearl, J. 2012. The Do-Calculus Revisited.

Qin, Y.; Wang, P.; and Li, C. 2021. The World is Binary: Contrastive Learning for Denoising Next Basket Recommendation. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 859–868. ACM.

Rendle, S.; Freudenthaler, C.; and Schmidt-Thieme, L. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*, 811–820.

Saito, Y.; Yaginuma, S.; Nishino, Y.; Sakata, H.; and Nakata, K. 2020. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, 501–509.

Shengyu, Z.; Yunze, T.; Kun, K.; Fuli, F.; Jiezhong, Q.; Jin, Y.; Zhou, Z.; Hongxia, Y.; Zhongfei, Z.; and Fei, W. 2023. Stable Prediction on Graphs with Agnostic Distribution Shifts. In *The KDD'23 Workshop on Causal Discovery, Prediction and Decision*, 49–74. PMLR.

Sun, L.; Bai, Y.; Du, B.; Liu, C.; Xiong, H.; and Lv, W. 2020. Dual Sequential Network for Temporal Sets Prediction. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1439–1448.

Wang, W.; Feng, F.; He, X.; Wang, X.; and Chua, T. 2021a. Deconfounded Recommendation for Alleviating Bias Amplification. In Zhu, F.; Ooi, B. C.; and Miao, C., eds., *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, 1717–1725. ACM.

Wang, W.; Feng, F.; He, X.; Zhang, H.; and Chua, T. 2021b. Clicks can be Cheating: Counterfactual Recommendation for Mitigating Clickbait Issue. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 1288–1297. ACM.

Wang, W.; Feng, F.; He, X.; Zhang, H.; and Chua, T.-S. 2021c. Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1288–1297.

Wang, Z.; Wei, W.; Cong, G.; Li, X.-L.; Mao, X.-L.; and Qiu, M. 2020. Global Context Enhanced Graph Neural Networks for Session-based Recommendation. In *Proceedings* *of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

Wei, T.; Feng, F.; Chen, J.; Wu, Z.; Yi, J.; and He, X. 2021. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In Zhu, F.; Ooi, B. C.; and Miao, C., eds., *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, 1791–1800. ACM.

Yu, L.; Liu, Z.; Zhu, T.; Sun, L.; Du, B.; and Lv, W. 2022a. Modelling Evolutionary and Stationary User Preferences for Temporal Sets Prediction. arXiv:2204.05490.

Yu, L.; Liu, Z.; Zhu, T.; Sun, L.; Du, B.; and Lv, W. 2023. Predicting temporal sets with simplified fully connected networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4835–4844.

Yu, L.; Sun, L.; Du, B.; Liu, C.; Xiong, H.; and Lv, W. 2020. Predicting Temporal Sets with Deep Neural Networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1083–1091.

Yu, L.; Wu, G.; Sun, L.; Du, B.; and Lv, W. 2022b. Element-guided Temporal Graph Representation Learning for Temporal Sets Prediction. In Laforest, F.; Troncy, R.; Simperl, E.; Agarwal, D.; Gionis, A.; Herman, I.; and Médini, L., eds., *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, 1902–1913. ACM.

Zhang, M.-L.; and Zhou, Z.-H. 2006. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE transactions on Knowledge and Data Engineering*, 18(10): 1338–1351.

Zhang, M.-L.; and Zhou, Z.-H. 2013. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8): 1819–1837.

Zhang, S.; Yao, D.; Zhao, Z.; Chua, T.; and Wu, F. 2021a. CauseRec: Counterfactual User Sequence Synthesis for Sequential Recommendation. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 367–377. ACM.

Zhang, S.; Yao, D.; Zhao, Z.; Chua, T.-S.; and Wu, F. 2021b. Causerec: Counterfactual user sequence synthesis for sequential recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 367–377.

Zhang, Y.; Feng, F.; He, X.; Wei, T.; Song, C.; Ling, G.; and Zhang, Y. 2021c. Causal intervention for leveraging popularity bias in recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 11–20.

Zhu, H.; Li, X.; Zhang, P.; Li, G.; He, J.; Li, H.; and Gai, K. 2018. Learning tree-based deep model for recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1079–1088.