Dual-Task Learning for Multi-Behavior Sequential Recommendation

Jinwei Luo
Shenzhen University
Shenzhen, China
luojinwei2016@email.szu.edu.cn

Mingkai He
Xiaolin Lin
Shenzhen University
Shenzhen, China

Weike Pan*
Shenzhen University
Shenzhen, China
panweike@szu.edu.cn

Zhong Ming*
Shenzhen University
Shenzhen, China
mingz@szu.edu.cn

ABSTRACT

Recently, sequential recommendation has become a research hotspot while multi-behavior sequential recommendation (MBSR) that exploits users' heterogeneous interactions in sequences has received relatively little attention. Existing works often overlook the complementary effect of different perspectives when addressing the MBSR problem. In addition, there are two specific challenges remained to be addressed. One is the heterogeneity of a user's intention and the context information, the other one is the sparsity of the interactions of target behavior. To release the potential of multi-behavior interaction sequences, we propose a novel framework named NextIP that adopts a dual-task learning strategy to convert the problem to two specific tasks, i.e., next-item prediction and purchase prediction. For next-item prediction, we design a target-behavior aware context aggregator (TBCG), which utilizes the next behavior to guide all kinds of behavior-specific item sub-sequences to jointly predict the next item. For purchase prediction, we design a behavior-aware self-attention (BSA) mechanism to extract a user's behavior-specific interests and treat them as negative samples to learn the user's purchase preferences. Extensive experimental results on two public datasets show that our NextIP performs significantly better than the state-of-the-art methods.

CCS CONCEPTS

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

KEYWORDS

Multi-Behavior Sequential Recommendation; Next-Item Prediction; Purchase Prediction; Self-Attention

ACM Reference Format:
Jinwei Luo, Mingkai He, Xiaolin Lin, Weike Pan, and Zhong Ming. 2022. Dual-Task Learning for Multi-Behavior Sequential Recommendation. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM '22), October 17–21, 2022, Atlanta, GA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3511808.3557298

*co-corresponding authors

CIKM '22, October 17–21, 2022, Atlanta, GA, USA

ACM ISBN 978-1-4503-9236-5/22/10…$15.00
https://doi.org/10.1145/3511808.3557298

1 INTRODUCTION

Recommender systems are often deemed indispensable for alleviating the information overload problem and assisting users to find their preferred items efficiently in many applications, e.g., e-commerce, news and entertainment [11]. Among the different research branches of recommender systems, sequential recommendation that aims to predict the next interaction (e.g., examination on an item) based on a user's previous interaction sequence of a certain type of behavior (e.g., examination) has become a research hotspot.

However, sequential recommendation that solely leverages the interaction data of one single type of behavior may result in unsatisfactory recommendation performance especially when the behavior data is sparse [22]. Taking the e-commerce scenario as an example, on one hand, a company's ultimate goal is to guide users to purchase items while the purchase data is usually very rare. On the other hand, a real-world e-commerce platform often contains some more abundant auxiliary behaviors (e.g., examinations, adds-to-cart and adds-to-favorite). Therefore, we have the potential to mine and transfer knowledge from the auxiliary behaviors to improve the performance in predicting the target behaviors, i.e., purchases. We name this type of research problem as multi-behavior sequential recommendation (MBSR) in this paper. Although there are so far very few studies on MBSR, we believe that MBSR is often more in line with the settings of various real-world scenarios, which thus deserves more attention.

The previous works on MBSR can be roughly classified into two branches: one is to model the behavior-agnostic item sequences and the corresponding behavior sequences [18, 29], and the other is to divide a multi-behavior interaction sequence into some behavior-specific item sub-sequences for later joint modeling [32]. In addition, modeling the transitions from some auxiliary behaviors to a target behavior is essential for understanding a user's target-oriented preferences. For example, a user may examine some different brands of earphones for comparison before making a purchase decision, where the transitions from previous examinations to a final purchase usually indicate the user's real purchase-oriented preferences.

Motivated by the above observations, we take a step forward and propose a novel solution named NextIP, which models a multi-behavior interaction sequence in a new perspective and models the transitions from some auxiliary behaviors to the target behavior from a user's perspective. Firstly, to unlock the potential of a multi-behavior interaction sequence, we treat it as a behavior sequence, some behavior-specific item sub-sequences, and a behavior-agnostic item sequence. Secondly, we utilize some item sequence encoders with different configurations to learn the item transition patterns at both the behavior-agnostic and behavior-specific levels.

Moreover, in order to address the intention heterogeneity issue and transfer different behavior-specific knowledge to varying degrees, we design a target-behavior aware context aggregator (TBCG) to exploit the next behavior to guide all kinds of behavior-specific item sub-sequences to jointly predict the next interaction. Thirdly, we introduce a second task that focuses on distinguishing the real purchase from the previous other auxiliary behaviors. Specifically, we design a behavior-aware self-attention (BSA) mechanism to extract a user's behavior-specific interests (i.e., virtual items) in a certain time interval and treat them as the negative samples to help learn the purchase preferences more effectively.

We summarize our main contributions as follows.

- We propose a novel solution named NextIP for an important and emerging problem, i.e., multi-behavior sequential recommendation (MBSR), which studies the problem from a new perspective by converting it to a next-item prediction task and a purchase prediction task.
- In the first task, we design a novel target-behavior aware context aggregator (TBCG) to achieve knowledge transfer among different sub-sequences, which helps predict the next item in a behavior-aware manner more accurately.
- In the second task, we explicitly model the heterogeneous transitions from a user's perspective via a novel behavior-aware self-attention (BSA) mechanism.
- We conduct extensive empirical studies on two real-world datasets and compare our NextIP with fifteen state-of-the-art methods, and show its effectiveness from five different angles.

2 RELATED WORK

In this section, we review some representative methods in single-behavior sequential recommendation, multi-behavior recommendation and multi-behavior sequential recommendation, respectively.

2.1 Single-Behavior Sequential Recommendation

The earliest works of sequential recommendation are based on Markov chains (MCs) and matrix factorization (MF) models. For instance, FPMC [26] models the long-term and short-term preferences of a user via MF and first-order MCs. Fossil [10] replaces the MF component in FPMC with the factored item similarity model (FISM) [15] and uses high-order MCs to consider more than one previous item. Besides, some researchers [9, 17] propose some translation-based methods for maintaining the triangle inequality in sequential data. With the development of deep learning, various techniques are applied to sequential recommendation. Recurrent neural network (RNN) is probably the very first technique due to its instincts of modeling sequential patterns [5, 13]. Convolutional neural network (CNN)-based methods such as Caser [28] and NextItNet [43] have also been proposed to consider skip connections and transitions at union-level. Based on the success of Transformer and the attention mechanism in NLP [30], SASRec [16] is proposed to use some self-attention modules to encode the sequential information of historical sequences. BERT4Rec [27] borrows the idea of the cloze task in NLP and trains a model in a bidirectional manner. FISSA [19] combines SASRec with a global preference learning model in a balanced way, which is found to be very effective for sequential recommendation.

In addition, many researchers propose to leverage graph neural network (GNN) to model the sequential data, especially in session-based recommendation [24, 33, 35, 40], where one important difference between session-based recommendation and sequential recommendation is that the data of the former is from anonymous users and the sequences are often relatively short [31].

2.2 Multi-Behavior Recommendation

Some works on multi-behavior recommendation (MBR) are based on MF models. For instance, adaptive BPR (ABPR) [23] learns the weight of different auxiliary behaviors adaptively and multi-channel BPR [20] proposes to sample some negative items from different behavior channels differently. Recently, deep learning-based methods show promising performance in MBR [2, 3, 37]. NMTR [6] extends NCF [11] to deal with a multi-behavior data with a multi-task framework. Moreover, many researchers propose to leverage GNN to capture the different relations between users and items. MBGCN [14] proposes to construct a (user, item) bipartite graph with multiple kinds of edges that represent different behaviors and uses graph convolutional network to enhance the representation learning. MB-GMN [38] further proposes a graph meta network to capture different behavior semantics and relations, which is found to be very effective in MBR. VAE++ [21] is the most recent variational autoencoder (VAE)-based method that simultaneously models three types of signals, including the target behavior, the auxiliary behavior and their mixed behaviors with multiple encoders and one single decoder. There are also some recent MBR methods that adopt contrastive learning to model the differences and correlations among different behavioral patterns of users [34, 36, 41], which is similar to our motivation in the purchase prediction task.

It is worth mentioning that there are some significant differences between MBR and MBSR, because capturing the complex sequential information (e.g., global/local preferences, behavior transitions, behavior tendency/continuity, long/short sequences, periodicity, etc.) is not a trivial task. This actually motivates numerous works on sequential recommendation (including a few studies on MBSR). Our empirical studies also show that capturing the sequential information is critical in delivering accurate recommendation services.

2.3 Multi-Behavior Sequential Recommendation

Though there are lots of works on multi-behavior recommendation [4], few of them pay attention to modeling the sequential dynamics in MBSR.

Existing works on MBSR can be divided into two categories according to the modeling perspectives. The first category is to model the behavior-agnostic item sequences and the corresponding behavior sequences [18, 22], which either uses the behavior sequence to enrich the input of the model or directly models the behavior sequences. For instance, RIB [44] concatenates a behavior embedding and an item embedding in the input layer of GRU. ASLI [29] uses a self-attention layer to model the item sequences, and a convolutional network to leverage the behavior and category sequences to learn a user's intents. MKM-SR [22] adopts a gated GNN (GGNN) to model the behavior-agnostic item sequence of a session and uses a GRU layer to model the corresponding behavior sequence to obtain the behavior representation at each step. The advantage of this type of methods is that they allow the models to distinguish different types of behaviors while retaining the integrity of an entire interaction sequence.

The second category is to divide a multi-behavior interaction sequence into some behavior-specific item sub-sequences and model them jointly [39, 42]. For example, MGNN-SPred [32] constructs a multi-relational item graph based on different behavior-specific item sequences for capturing the examination-to-examination and purchase-to-purchase relationships. DMT [7] proposes a deep interest Transformer to model each behavior-specific item sequence for the subsequent CTR task and CVR task. The merit of this modeling perspective is that the problem can be simplified into modeling the item sequences since the semantics of each behavior type can be learned via different parameters.

Different from the above approaches, we propose a new modeling perspective for MBSR that leverages all the behavior sequences, behavior-agnostic and behavior-specific item sequences in a unified and complementary way. Moreover, we address the heterogeneity challenge and the sparsity challenge well in the next-item prediction task and purchase prediction task, respectively. Note that though MKM-SR [22] also adopts multi-task learning by introducing an auxiliary task of learning knowledge embeddings from a knowledge graph, our NextIP is different from it by converting the MBSR problem to two specific tasks, i.e., next-item prediction and purchase prediction.

3 PRELIMINARIES

In this section, we first formally define the problem and give an overview of the proposed method as shown in Figure 1.

3.1 Problem Definition

In a real-world platform such as an e-commerce site, we have a set of users $\mathcal{U} = \{u\}$ and a set of items $\mathcal{I} = \{i\}$, where users' behaviors to items can be of multiple types $\mathcal{B} = \{e, f, c, p\}$ such as examination (e), add-to-favorite (f), add-to-cart (c) and purchase (p). Each user u is associated with an interaction sequence of (item, behavior) pairs $\mathcal{S}_u = \left\{\left(i_u^1, b_u^1\right), \ldots, \left(i_u^\ell, b_u^\ell\right), \ldots, \left(i_u^L, b_u^L\right)\right\}$, where $i_u^\ell \in \mathcal{I}$ denotes the ℓth item interacted by user u with behavior $b_u^\ell \in \mathcal{B}$.

The goal of multi-behavior sequential recommendation (MBSR) is then to exploit the information in $\mathcal{S}_u$ and recommend the next likely-to-purchase item for each user u.

3.2 Challenges

For the studied MBSR problem, there are two specific challenges that remained to be addressed. From the perspective of sequence, users often have different behavioral intentions (i.e., examination, purchase) at different time steps, and there are different contextual information of behaviors in the historical process. Therefore, the first challenge is the heterogeneity of a user's intention and the context information, i.e., how to adaptively utilize heterogeneous contextual information for a user with different behavioral intentions. From the perspective of behavior, the second challenge lies in the sparsity of the data of target behavior, hindering us to learn a user's purchase preferences effectively.

Table 1: Some notations and their explanations used in the paper.

$\mathcal{U} = \{u\}$	the whole set of users
$\mathcal{I} = \{i\}$	the whole set of items
$\mathcal{B}$	the set of behaviors
$i_u^\ell \in \mathcal{I}$	the ℓth item interacted by user u
$b_u^\ell \in \mathcal{B}$	the ℓth behavior type interacted by user u
$\mathcal{S}_u = \{(i_u^\ell, b_u^\ell)\}$	the interaction sequence of user u
$d \in \mathbb{R}$	number of latent dimensions
$L \in \mathbb{R}$	length of the sequence
$\boldsymbol{U} \in \mathbb{R}^{\lvert\mathcal{U}\rvert \times d}$	user embedding matrix
$\boldsymbol{u}_u \in \mathbb{R}^{1\times d}$	user embedding of user u
$\boldsymbol{H} \in \mathbb{R}^{\lvert\mathcal{I}\rvert \times d}$	item embedding matrix
$\boldsymbol{h}_i \in \mathbb{R}^{1\times d}$	item embedding of item i
$\boldsymbol{B} \in \mathbb{R}^{\lvert\mathcal{B}\rvert \times d}$	behavior embedding matrix
$\boldsymbol{b}_{b_u^\ell}$	behavior embedding of user u at step ℓ
$\boldsymbol{P} \in \mathbb{R}^{L\times d}$	position embedding matrix
$SAB(\cdot)$	self-attention block
$\boldsymbol{X}_u^{(k)} \in \mathbb{R}^{L\times d}$	representation matrix of a behavior-agnostic item sequence in kth layer of SAB
$\boldsymbol{X}_u^{e(k)} \in \mathbb{R}^{L\times d}$	representation matrix of an examination-specific item sequence in kth layer of SAB
$\boldsymbol{x}_{u,\ell} \in \mathbb{R}^{1\times d}$	behavior-agnostic sequence representation of user u at step ℓ
$\boldsymbol{g}'_{u,\ell} \in \mathbb{R}^{1\times d}$	behavior-aware context representation of user u at step ℓ

3.3 Overview

As a response to the aforementioned two challenges, we propose a novel and effective framework named NextIP that adopts a dual-task learning strategy to convert the problem to two specific tasks, i.e., next-item prediction and purchase prediction. Firstly, we address the heterogeneity challenge in task 1 (in the bottom left corner of Figure 1) by designing a novel target-behavior aware context aggregator (TBCG) to transfer the unique knowledge of different types of behaviors so as to predict the next item in a behavior-aware manner. Secondly, we address the sparsity challenge in task 2 (in the bottom right corner of Figure 1). Specifically, all the items that a user has interacted with are treated as positive samples in task 1, while in task 2, we propose to treat the items associated with auxiliary behaviors as negative signals to refine the learning of a user's purchase-oriented preferences.

4 PROPOSED METHODOLOGY

In this section, we introduce each component of our NextIP in detail. We list some notions and their explanations used in this paper in Table 1.

4.1 Embedding Layer

First of all, we present an embedding layer to convert an input multi-behavior interaction sequence $\mathcal{S}_u$ to different embeddings.

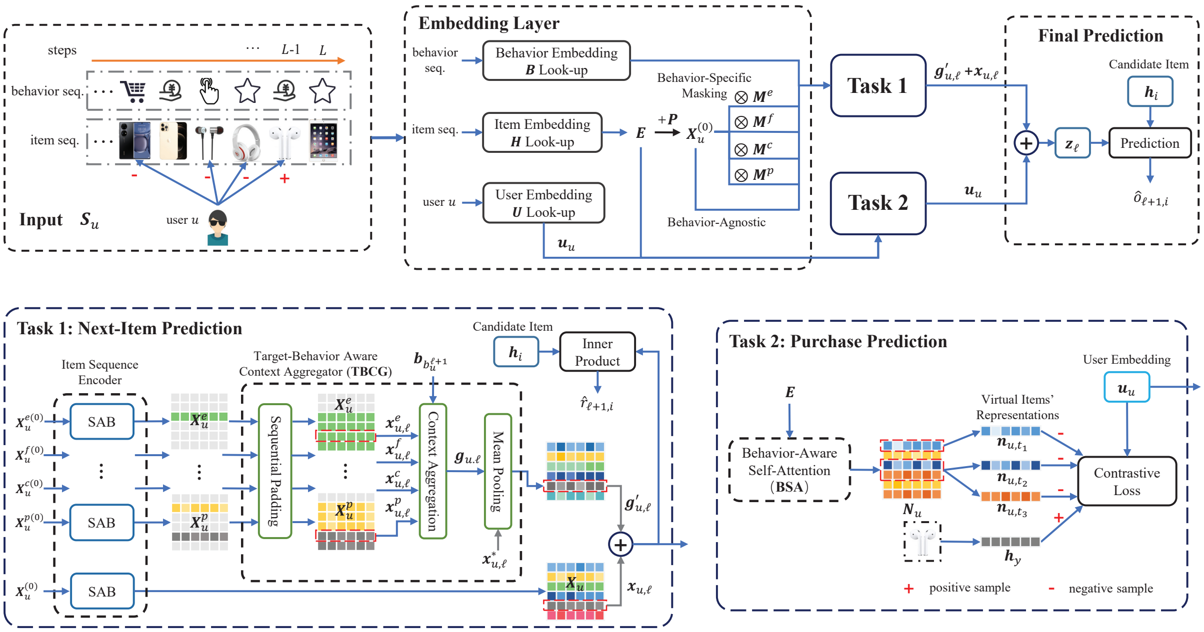


Figure 1: The overview of our proposed NextIP. The pipeline of our NextIP is illustrated in the upper half part. In general, our NextIP takes a multi-behavior interaction sequence of user u as input and obtains the representations of the behavior sequence, each behavior-specific item sub-sequence and the behavior-agnostic item sequence as well as the user embedding in the embedding layer. These representations are then fed into two well-designed tasks, i.e., a next-item prediction task (in the bottom left corner) and a purchase prediction task (in the bottom right corner) to capture both the user's behavior-aware short-term interests and purchase-oriented long-term preferences, which are finally used to fulfill the final prediction.

We have a learnable behavior embedding matrix $\boldsymbol{B} \in \mathbb{R}^{|\mathcal{B}|\times d}$, an item embedding matrix $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{I}|\times d}$, a position embedding matrix $\boldsymbol{P} \in \mathbb{R}^{|L|\times d}$ and a user embedding matrix $\boldsymbol{U} \in \mathbb{R}^{|\mathcal{U}|\times d}$, where d is the size of latent dimension. Through embedding look-up operations from $\boldsymbol{H}$, we can retrieve and stack the item embeddings of $\mathcal{S}_u$ as an embedding matrix $\boldsymbol{E} = \left[\boldsymbol{h}_{i_u^1};\ldots;\boldsymbol{h}_{i_u^\ell};\ldots;\boldsymbol{h}_{i_u^L}\right] \in \mathbb{R}^{L\times d}$. Following [16], the representation of a behavior-agnostic item sequence can be obtained by adding two embedding matrices, i.e., $\boldsymbol{X}_u^{(0)} = \boldsymbol{E} + \boldsymbol{P}$.

In order to unlock the potential of a multi-behavior interaction sequence, we further represent a user's behavior-specific item sequence by behavior-specific masking. Without loss of generality, we take the examination-specific item sequence as an example,

$$\boldsymbol{X}_u^{e(0)} = \boldsymbol{X}_u^{(0)} \otimes \boldsymbol{M}^e, \tag{1}$$

where $\otimes$ is the element-wise product. $\boldsymbol{M}^e = [\boldsymbol{m}_1^e;\ldots;\boldsymbol{m}_\ell^e;\ldots;\boldsymbol{m}_L^e] \in \mathbb{R}^{L\times d}$ denotes the examination-specific mask matrix, in which $\boldsymbol{m}_\ell^e$ is a vector of ones if $b_u^\ell = e$, and a vector of zeros otherwise. Similarly, we can obtain $\boldsymbol{X}_u^{f(0)}, \boldsymbol{X}_u^{c(0)}$ and $\boldsymbol{X}_u^{p(0)}$ as the embedding matrices of the other three behavior-specific item sequences.

4.2 Task 1: Next-Item Prediction

The first task is to predict the next item in a behavior-aware manner by utilizing the behavior sequences, behavior-specific and behavior-agnostic item sequences in a unified and complementary way. In this task, we first encode the behavior-agnostic preferences and behavior-specific context in an item sequence encoder and then introduce a novel target-behavior aware context aggregator (TBCG) to transfer the unique knowledge of different types of behaviors.

4.2.1 Item Sequence Encoder. To obtain different context information in all the behavior-agnostic and behavior-specific sequences, we choose self-attention block (SAB) [16] for the item sequence encoder since it is known effective. Moreover, the parameters of each self-attention network are independent across different kinds of item sequences, which are used to capture different semantic information and transition patterns. For instance, the transitions in an examination-specific item sequence usually indicate similarity among the items while those in a purchase-specific item sequence often mean complementarity to some extent.

Omitting the formulas of the layer normalization and residual connection, each SAB consists of a self-attention layer (SAL), and a feed-forward layer (FFL),

$$SAB(\boldsymbol{X}) = FFL(SAL(\boldsymbol{X})), \tag{2}$$

$$\boldsymbol{X}' = SAL(\boldsymbol{X}) = (softmax(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}) \otimes \Delta)\boldsymbol{V}, \tag{3}$$

$$FFL(\boldsymbol{X}') = ReLU(\boldsymbol{X}'\boldsymbol{W}_1 + \boldsymbol{b}_1)\boldsymbol{W}_2 + \boldsymbol{b}_2, \tag{4}$$

where $\boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}_Q$, $\boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}_K$ and $\boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}_V$ with $\boldsymbol{W}_Q, \boldsymbol{W}_K, \boldsymbol{W}_V \in \mathbb{R}^{d\times d}$ are the projected query, key and value matrices, respectively. Note that $\sqrt{d}$ in SAL($\cdot$) is the scaling factor to prevent overlarge values of the inner product and Δ is the causality mask used to ensure that only the previous ℓ items are taken into account when predicting the $(\ell+1)$th item. $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathbb{R}^{d\times d}$ and $\boldsymbol{b}_1, \boldsymbol{b}_2 \in \mathbb{R}^{1\times d}$ are learnable weights and biases of the two-layer network, respectively.

By feeding $\boldsymbol{X}_u^{(0)}$ into the corresponding K layers of SABs, we can obtain $\boldsymbol{X}_u^{(K)} = [\boldsymbol{x}_{u,1}^{(K)}; \ldots; \boldsymbol{x}_{u,\ell}^{(K)}; \ldots; \boldsymbol{x}_{u,L}^{(K)}]$, where $\boldsymbol{x}_{u,\ell}^{(K)} \in \mathbb{R}^{1\times d}$ denotes the representation of the user's behavior-agnostic context at step ℓ. Similarly, we can obtain $\boldsymbol{X}_u^{e(K)}, \boldsymbol{X}_u^{f(K)}, \boldsymbol{X}_u^{c(K)}, \boldsymbol{X}_u^{p(K)}$ for the behavior-specific item sequences. Note that we remove the superscript K for brevity in Figure 1 and in the subsequent text.

4.2.2 Target-Behavior Aware Context Aggregator. Different types of behavior-specific item sequences can provide different contextual information, which thus should be explicitly exploited. For instance, the records of a user's recent examinations on some earphones are useful signals that the next purchase might be an earphone, and knowing a user bought an iPhone last month can inform us to predict that the user is likely to buy an AirPods.

However, the context information needs to be well balanced when predicting different behaviors, since the dependency across different behaviors is complex. Therefore, we design a target-behavior aware context aggregator (TBCG), which takes a user's target behavior embedding to improve the context aggregation. Specifically, at the current step ℓ, we stack up the context representations of different behaviors (i.e., $[\boldsymbol{x}_{u,\ell}^e; \boldsymbol{x}_{u,\ell}^f; \boldsymbol{x}_{u,\ell}^c; \boldsymbol{x}_{u,\ell}^p]$) and take them as keys and values, while leveraging the user's behavior embedding at the next step $\boldsymbol{b}_{b_u^{\ell+1}}$ as a query to adaptively extract the context that is more relevant to predict the next item,

$$\boldsymbol{Q}'_\ell = \boldsymbol{b}_{b_u^{\ell+1}} \boldsymbol{W}_{Q'}, \tag{5}$$

$$\boldsymbol{K}'_\ell = [\boldsymbol{x}_{u,\ell}^e; \boldsymbol{x}_{u,\ell}^f; \boldsymbol{x}_{u,\ell}^c; \boldsymbol{x}_{u,\ell}^p] \boldsymbol{W}_{K'}, \tag{6}$$

$$\boldsymbol{V}'_\ell = [\boldsymbol{x}_{u,\ell}^e; \boldsymbol{x}_{u,\ell}^f; \boldsymbol{x}_{u,\ell}^c; \boldsymbol{x}_{u,\ell}^p] \boldsymbol{W}_{V'}, \tag{7}$$

$$\boldsymbol{g}_{u,\ell} = softmax(\frac{\boldsymbol{Q}'_\ell {\boldsymbol{K}'_\ell}^T}{\sqrt{d}}) \boldsymbol{V}'_\ell, \tag{8}$$

where $\boldsymbol{Q}'_\ell \in \mathbb{R}^{1\times d}$, $\boldsymbol{K}'_\ell \in \mathbb{R}^{4\times d}$ and $\boldsymbol{V}'_\ell \in \mathbb{R}^{4\times d}$ are the projected query, key and value matrices, respectively. And $\boldsymbol{W}_{Q'}, \boldsymbol{W}_{K'}, \boldsymbol{W}_{V'} \in \mathbb{R}^{d\times d}$ are the learnable weight matrices. Note that $\boldsymbol{g}_{u,\ell}$ denotes the gated context representation of user u at step ℓ considering the user's target behavior.

After that, we can further obtain the refined behavior-aware context representation by combining $\boldsymbol{g}_{u,\ell}$ with the context representation of user u at step ℓ for behavior type $*$, i.e., $\boldsymbol{x}_{u,\ell}^*$, via mean pooling,

$$\boldsymbol{g}'_{u,\ell} = \frac{\boldsymbol{g}_{u,\ell} + \boldsymbol{x}_{u,\ell}^*}{2}, \tag{9}$$

where $*$ denotes the next behavior of the current step, i.e., $b_u^{\ell+1} \in \mathcal{B}$. We find that setting the weight of the elements uniformly often leads to good performance in our empirical studies. The main idea behind Eq.(9) is to enhance the contribution of the context representation of the next behavior type at the current step in the training phase. Note that in the evaluation phase, we will fix the next behavior type as purchase p, since our goal is next purchased item prediction.

Notably, to facilitate matrix-level operations, we utilize sequential padding to fill the empty values of each behavior-specific context matrix to ensure that the vector at step ℓ corresponds to the behavior-specific context information of the most recent step. Finally, we predict the probability that user u will interact with item i at the $(\ell+1)$th step as follows,

$$\hat{r}_{\ell+1,i} = (\boldsymbol{x}_{u,\ell} + \boldsymbol{g}'_{u,\ell})(\boldsymbol{h}_i)^T. \tag{10}$$

We adopt the typical binary cross-entropy loss for task 1, which is the same as that in the seminal method SASRec [16],

$$\mathcal{L}_1 = -\sum_{u\in\mathcal{U}} \sum_{\ell=2}^{L+1} \delta(i_u^\ell)[\log(\sigma(\hat{r}_{\ell,i_u^\ell})) + \log(1-\sigma(\hat{r}_{\ell,j}))], \tag{11}$$

where j is a negative item randomly sampled from $\mathcal{I}\backslash\mathcal{S}_u$ for each position $\ell \in \{2, ..., L+1\}$. The indicator function $\delta(i_u^\ell) = 1$ if i_u^ℓ is not a padding item, and 0 otherwise. The indicator function $\delta(i_u^\ell)$ is used to ignore the loss value when i_u^ℓ is a padding item, since we follow the common practice and use a padding item to pad the user sequence to a same length L.

4.3 Task 2: Purchase Prediction

Task 1 is designed to predict the next item of all behaviors and to learn a user's short-term interests while task 2 aims to learn a user's purchase-oriented long-term preferences. In addition, a user's purchase behavior is often much sparser in his or her interaction history, hindering us to learn the user's purchase preferences effectively.

To address the above issues, we leverage a user's behavior transitions to learn his or her purchase-oriented preferences in a self-supervised manner. For instance, users often interact with different brands of earphones for comparison via some auxiliary behaviors (e.g., examination) before making a purchase decision. If a user finally chooses to purchase an item instead of other interacted items, this contains fine-grained information about the user's purchase preferences. Therefore, we should pay attention to distinguishing the purchased item from the previous items with some auxiliary behaviors.

4.3.1 Behavior-Aware Self-Attention. The intentions reflected in a user's interaction history are usually diverse and uncertain. For example, a user might examine clothes before purchasing an earphone, and thus assuming the user prefers the earphone to each cloth may not be reasonable. To alleviate this uncertainty issue, we design a behavior-aware self-attention (BSA) mechanism to aggregate the collection of the historically interacted items of each kind of behavior into a virtual item w.r.t. each step. Firstly, we create an attention mask as follows,

$$\boldsymbol{A} = \boldsymbol{M} \otimes \Delta, \tag{12}$$

where $\boldsymbol{M} \in \mathbb{R}^{L\times L}$ denotes the behavior-aware attention mask, $\boldsymbol{M}_{\ell,\ell'} = 1$ if $b_u^\ell = b_u^{\ell'}$ and $\ell - \ell' \le L_{clip}$, and $\boldsymbol{M}_{\ell,\ell'} = 0$ otherwise. The usage of a length clip is to ensure that the aggregated items correspond to a similar intention of user u. In order to further illustrate the attention mask in our NextIP shown in Eq.(12), we give an example in Figure 2. Next, we can use self-attention to aggregate each

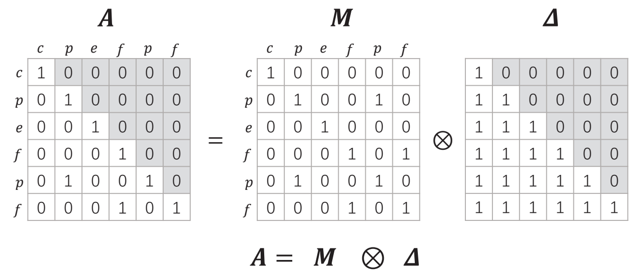


Figure 2: An example of the attention mask in our NextIP.

behavior-specific item set into a virtual item representation,

$$\boldsymbol{N}_u = (softmax(\frac{\boldsymbol{Q}''\boldsymbol{K}''^T}{\sqrt{d}}) \otimes \boldsymbol{A})\boldsymbol{V}'', \tag{13}$$

where $\boldsymbol{Q}'' = \boldsymbol{E}\boldsymbol{W}_{Q''}$, $\boldsymbol{K}'' = \boldsymbol{E}\boldsymbol{W}_{K''}$ and $\boldsymbol{V}'' = \boldsymbol{E}\boldsymbol{W}_{V''}$ with $\boldsymbol{W}_{Q''}$, $\boldsymbol{W}_{K''}$, $\boldsymbol{W}_{V''} \in \mathbb{R}^{d\times d}$ as the projected query, key and value matrices, respectively. After BSA, for user u, we can obtain $\boldsymbol{N}_u = [\boldsymbol{n}_{u,1}; \ldots; \boldsymbol{n}_{u,\ell}; \ldots; \boldsymbol{n}_{u,L}]$, where $\boldsymbol{n}_{u,\ell} \in \mathbb{R}^{1\times d}$ denotes the representation of a virtual item that user u has recently interacted with using a certain behavior at step ℓ.

We then treat the virtual items' representations of every auxiliary behavior (e.g., examination) before a purchased item as negative samples, which will be used to learn the user's purchase-oriented preferences. We adopt the contrastive loss function to train the model as follows,

$$g(\boldsymbol{h}_y) = \exp\left(\boldsymbol{u}_u \boldsymbol{W} \boldsymbol{h}_y^T / \rho\right), \tag{14}$$

$$g(\boldsymbol{n}_{u,t}) = \exp\left(\boldsymbol{u}_u \boldsymbol{W} \boldsymbol{n}_{u,t}^T / \rho\right), \tag{15}$$

$$\mathcal{L}_2 = -\sum_{u\in\mathcal{U}} \sum_{y\in\mathcal{Y}_u} \log \frac{g(\boldsymbol{h}_y)}{g(\boldsymbol{h}_y) + \sum_{t\in\mathcal{T}(u,y)} g(\boldsymbol{n}_{u,t})}, \tag{16}$$

where $y \in \mathcal{Y}_u$ is an item purchased by user u, $\boldsymbol{u}_u \in \mathbb{R}^{1\times d}$ denotes the user embedding, and ρ is the temperature parameter. $\boldsymbol{W}$ is a learnable weight matrix to perform feature transformation, and $\mathcal{T}(u, y)$ is a set containing the step indices of each type of auxiliary behavior that is closest to the user's purchased item y. Apart from the contrastive loss in Eq.(16), we can also use a BPR loss to distinguish positive samples from negative ones.

4.4 Learning and Final Prediction

We train our NextIP in an end-to-end fashion by minimizing the loss on both task 1 and task 2,

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2. \tag{17}$$

Note that simply setting the weight of $\mathcal{L}_2$ to 1 often leads to good performance in our empirical studies.

Finally, we can predict the probability that user u will purchase item i at the $(\ell + 1)$th step as follows,

$$\hat{o}_{\ell+1,i} = \boldsymbol{z}_\ell (\boldsymbol{h}_i)^T. \tag{18}$$

where $\boldsymbol{z}_\ell = \boldsymbol{x}_{u,\ell} + \boldsymbol{g}'_{u,\ell} + \boldsymbol{u}_u$. Note that $\boldsymbol{x}_{u,\ell}$ and $\boldsymbol{g}'_{u,\ell}$ are optimized in $\mathcal{L}_1$ while $\boldsymbol{u}_u$ is optimized in $\mathcal{L}_2$.

Dataset	# Users	# Items	Avg. Length	Behavior set
UB	20,858	30,853	33.71	$\{e, f, c, p\}$
Tmall	17,209	16,174	48.60	$\{e, f, p\}$

Table 2: Statistics of the processed datasets, where Avg. Length denote the average length of users' interaction sequences in the datasets, and e, f, c and p denote examination, add-to-favorite, add-to-cart and purchase, respectively.

5 EXPERIMENTS

In this section, we first introduce the datasets and the experimental settings. Then, we conduct extensive empirical studies to investigate the following four research questions:

- RQ1: How does our NextIP perform when competing with the state-of-the-art baselines?
- RQ2: What is the impact of different components in our NextIP?
- RQ3: What is the impact of the different behavior-specific item sub-sequences in our NextIP?
- RQ4: What is the rationality of our NextIP? How important are different source behaviors in predicting different behaviors?

5.1 Datasets and Evaluation Metrics

We conduct offline experiments (online training and inference is an important future direction) on two public and real-world datasets in e-commerce scenarios, i.e., Tmall[1] and User Behavior (UB)[2], which are released at the IJCAI Competitions 2015 and 2016, respectively. Both datasets contain different types of behaviors, i.e., examination, add-to-favorite, add-to-cart and purchase. We preprocess the datasets as follows: (i) for duplicated (user, item, behavior) tuples in a sequence, we only retain the first one; (ii) we discard the cold-start items with fewer than 10 and 20 purchase interactions for UB and Tmall, respectively; (iii) we discard the cold-start users with fewer than 5 and 10 purchase interactions for UB and Tmall, respectively; (iv) we remove the records of adds-to-cart in Tmall because of its rarity; (v) for each user, we take the last two purchase interactions as the validation and test data (note that the interactions between them are kept in final evaluation), and those before the penultimate purchase as the training data; and (vi) for the preferred items in the test data of each user, we remove them from the training data since we aim to recommend new items [32]. The statistics of the processed datasets are summarized in Table 2.

We evaluate the recommendation performance via recall (Rec@N) and normalized discounted cumulative gain (NDCG@N), where $N \in \{1, 5, 10\}$. Rec@N means the proportion of cases when the preferred item is in a top-N recommendation list, while NDCG@N pays attention to whether it has a relatively high-ranking position. We follow [16, 19] and prepare a candidate list with 100 randomly sampled un-interacted items according to the item popularity for each user. Note that we also report the experimental results under the full-ranking setting, in which all items are considered as candidates without sampling.

[1] https://tianchi.aliyun.com/dataset/dataDetail?dataId=42
[2] https://tianchi.aliyun.com/dataset/dataDetail?dataId=649

5.2 Baselines and Parameter Configurations

To show the effectiveness of our NextIP, we compare it with four groups of baselines. The first group contains two classic single-behavior recommendation (SBR) methods:

BPRMF [25]. A non-sequential model that optimizes matrix factorization using a pairwise ranking loss.

FISM [15]. A non-sequential model that represents a user with his or her interacted items.

The second group contains two state-of-the-art multi-behavior recommendation (MBR) methods:

MB-GMN [38]. A state-of-the-art GNN-based multi-behavior model that incorporates a graph meta network to learn multi-behavior patterns.

VAE++ [21]. A state-of-the-art VAE-based multi-behavior model that utilizes three types of signals, including the purchases, the examinations and their mixed behaviors.

The third group contains six single-behavior sequential recommendation (SBSR) methods:

FPMC [26]. A classic sequential model based on matrix factorization and first-order Markov chains (MCs).

Fossil [10]. A classic sequential model which combines FISM [15] and high-order MCs to consider more than one previous item.

GRU4Rec+ [12]. An RNN-based model which improves GRU4Rec [13] by applying a BPR-max loss and an additional sampling strategy.

Caser [28]. A CNN-based model that utilizes some horizontal and vertical convolutional filters to capture different sequential patterns.

SASRec [16]. A pioneering model based on hierarchical self-attention modules.

FISSA [19]. A recent model for sequential recommendation that uses SASRec to learn a user's local preferences, an attentive version of FISM to learn his or her global preferences, and a gating module for balancing these two parts.

The fourth group contains five multi-behavior sequential recommendation (MBSR) methods:

RIB [44]. An RNN-based model that takes the concatenation of the item embedding and the behavior embedding as the input of a GRU layer.

BINN [18]. An RNN-based model that designs a contextual long short-term memory (CLSTM) structure to model the item and behavior sequences.

MGNN-SPred [32]. A GNN-based model for MBSR that constructs a multi-relational item graph based on all kinds of behavior-specific item sub-sequences.

M-SR [22]. A GNN-based model that uses a gated GNN (GGNN) to model the item sequences and uses a GRU layer to model the sequential patterns from sequences of operations (i.e., behavior types). Note that M-SR is a reduced version of MKM-SR without the knowledge graph.

ASLI [29]. A recent model that uses a self-attention layer to model the item sequences, and a convolutional network to leverage the behavior and category sequences to obtain the users' intents, which are then used to query the relevant items. Note that we do not use the category sequences in the studied problem in this paper.

For RIB, BINN and ASLI, we implement them by TensorFlow. For other methods, we use the code released by their authors[3,4,5,6,7,8,9] [10]. For fair comparison, we fix the embedding dimension d of tune all models as 50 and all the hyper-parameters on the validation data following the suggestions in the original papers. For our NextIP, following [16, 19], we set the sequence length L to 50, the batch size to 128, the dropout rate to 0.5, and adopt the Adam optimizer with a learning rate of 0.001. The length clip L_{clip} in BSA is set to 10 and the temperature parameter ρ of the contrastive loss is set to 0.07 [8]. The number of self-attention layers for all kinds of item sequences are searched from $K \in \{1, 2, 3\}$ [16, 19].

5.3 Results

5.3.1 Main Results (RQ1). We report the experimental results of fifteen baselines and our NextIP in Table 3. To further verify the effectiveness of our NextIP, we also report the experimental results of four representative baselines from four different groups and our NextIP under the full-ranking setting (i.e., all items are candidates) in Table 4. The best one of each column is marked in bold and the second best one is underlined.

For non-sequential single-behavior recommendation methods, FISM beats BPRMF in most cases, which indicates that modeling of user representation from all the interacted items can often lead to performance improvement.

For non-sequential multi-behavior recommendation methods, both the GNN-based method MB-GMN and VAE-based method VAE++ outperform the single-behavior recommendation methods in all cases, indicating the benefits of mining multi-behavior preferences. However, the performance of these two state-of-the-art MBR methods is worse than deep-learning-based SBSR methods and MBSR methods. It indicates that capturing the complex sequential information (e.g., global or local preferences, behavior transitions, behavior dependency, etc.) is critical in sparse sequence data.

Among the methods for sequential recommendation, we can have the following observations.

- These methods surpass the non-sequential methods to a large extent in all cases, showing the importance of modeling the sequential information.
- SASRec and FISSA consistently perform better than Caser and GRU4Rec+ on both datasets, which is consistent with the observations in previous studies [16, 19] and indicates the advantage of a self-attention network for modeling the item sequences.

Among the methods for multi-behavior sequential recommendation, we can observe the following phenomena.

- The RNN-based methods, i.e., RIB and BINN, beat the single-behavior sequential method GRU4Rec+, which showcases that their modeling of the behavior sequences to distinguish different preferences of users benefits the performance.

[3]https://cseweb.ucsd.edu/~jmcauley/
[4]https://github.com/hidasib/GRU4Rec
[5]https://github.com/graytowne/caser_pytorch
[6]http://csse.szu.edu.cn/staff/panwk/publications/FISSA/
[7]https://github.com/ciecus/MKM-SR
[8]https://github.com/Autumn945/MGNN-SPred
[9]https://github.com/akaxlh/MB-GMN
[10]https://csse.szu.edu.cn/staff/panwk/publications/VAEplusplus/

Model	UB					Tmall				
	Rec@1	Rec@5	NDCG@5	Rec@10	NDCG@10	Rec@1	Rec@5	NDCG@5	Rec@10	NDCG@10
BPRMF	0.086	0.211	0.149	0.309	0.181	0.050	0.166	0.108	0.266	0.140
FISM	0.095	0.246	0.172	0.362	0.209	0.046	0.171	0.109	0.273	0.142
MB-GMN	0.094	0.251	0.175	0.364	0.208	0.061	0.211	0.136	0.342	0.181
VAE++	0.139	0.290	0.215	0.398	0.250	0.088	0.238	0.167	0.360	0.203
FPMC	0.104	0.257	0.182	0.372	0.219	0.112	0.277	0.196	0.394	0.234
Fossil	0.085	0.219	0.153	0.319	0.185	0.094	0.244	0.169	0.356	0.206
GRU4Rec+	0.225	0.367	0.300	0.451	0.327	0.195	0.381	0.291	0.493	0.327
Caser	0.187	0.353	0.274	0.444	0.304	0.161	0.384	0.275	0.514	0.317
SASRec	0.226	0.446	0.341	0.556	0.376	0.210	0.481	0.351	0.616	0.395
FISSA	0.224	0.493	0.364	0.622	0.405	0.195	0.484	0.344	0.634	0.392
RIB	0.214	0.390	0.306	0.488	0.337	0.205	0.425	0.319	0.547	0.359
BINN	0.223	0.402	0.316	0.505	0.349	0.223	0.434	0.332	0.552	0.370
MGNN-SPred	0.146	0.291	0.220	0.392	0.253	0.165	0.391	0.282	0.521	0.324
M-SR	0.224	0.401	0.316	0.500	0.348	0.217	0.426	0.325	0.547	0.365
ASLI	0.230	0.452	0.347	0.562	0.382	0.215	0.490	0.359	0.623	0.402
NextIP	**0.247**	**0.509**	**0.384**	**0.632**	**0.423**	**0.246**	**0.548**	**0.403**	**0.681**	**0.446**

Table 3: Recommendation performance of our NextIP and four groups of baselines on UB and Tmall. Note that the best one of each column is marked in bold, and the second best result is underlined.

Dataset	Metric	BPRMF	VAE++	SASRec	ASLI	NextIP
UB	Rec@5	0.0143	0.0377	0.0436	0.0423	**0.0448**
	NDCG@5	0.0086	0.0250	0.0224	0.0221	**0.0231**
	Rec@10	0.0281	0.0564	0.0766	0.0731	**0.0790**
	NDCG@10	0.0130	0.0310	0.0331	0.0320	**0.0340**
Tmall	Rec@5	0.0094	0.0255	0.0488	0.0514	**0.0542**
	NDCG@5	0.0057	0.0175	0.0271	0.0283	**0.0314**
	Rec@10	0.0189	0.0387	0.0821	0.0859	**0.0896**
	NDCG@10	0.0087	0.0217	0.0379	0.0394	**0.0428**

Table 4: Recommendation performance of our NextIP and four representative baselines on UB and Tmall under the full-ranking setting, in which all items are considered as candidates. Note that the best one of each column is marked in bold, and the second best result is underlined.

- MGNN-SPred performs poorly in our setting, which is believed because of uncompetitiveness of GNN-based methods for sequential recommendation. This is also observed by other researchers [1].
- ASLI outperforms BINN on both datasets, especially when coping with longer sequences in Tmall. The reason is that ASLI introduces an additional convolution module and an auxiliary loss to force the model to predict the next latent intent (i.e., the behavior type in our case), and the self-attention network is more suitable for modeling long item sequences than RNN.

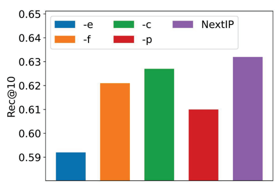
The experimental results in Table 4 shows the same tendency as that in Table 3. Our NextIP consistently achieves the best performance on both datasets comparing with all the baselines, which clearly demonstrates its superiority in modeling users' sequential and heterogeneous behaviors. Different from all the existing MBSR methods, our NextIP adopts a dual-task learning strategy to utilize the behavior sequences, behavior-specific and behavior-agnostic item sequences in a novel and unified way. In addition, we address the heterogeneity challenge in task 1 and the sparsity challenge in task 2, leading to significantly better performance.

5.3.2 Ablation Study (RQ2). In order to understand the contribution of different components to the performance of our NextIP, we conduct an ablation study and report the results in Table 5. We have the following observations.

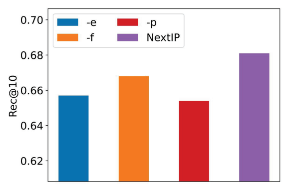
- By removing the target-behavior aware context aggregator (TBCG) in task 1 and the behavior-aware self-attention (BSA) mechanism in task 2, our NextIP reduces to SASRec, i.e., NextIP(w/o TBCG&BSA), which only contains task 1 and utilizes the behavior-agnostic item sequences. The improvement of NextIP(w/o TBCG) over NextIP(w/o TBCG&BSA) demonstrates the rationality of constructing the purchase prediction task to exploit the heterogeneous transitions from auxiliary behaviors to target behaviors in a user's perspective.
- The performance gap between NextIP(w/o TBCG&BSA) and NextIP(w/o BSA) indicates that transferring the unique knowledge of different types of behaviors to help predict the next interaction in a behavior-aware manner can improve the performance.
- Comparing NextIP(w/o BSA) with NextIP(w/o BSA & $\boldsymbol{g}_{u,\ell}$ in TBCG), i.e., the variant of NextIP(w/o BSA) that removes $\boldsymbol{g}_{u,\ell}$ in TBCG and only uses $\boldsymbol{x}^*_{u,\ell}$ in Eq.(9), the performance decreases showing the usefulness of the target behavior embedding in balancing the context information from different behaviors in our TBCG. Similarly, we can see the necessity of using $\boldsymbol{x}^*_{u,\ell}$ to refine the final context representation by comparing NextIP(w/o BSA) with NextIP(w/o BSA & $\boldsymbol{x}^*_{u,\ell}$ in TBCG).
- NextIP(w/o BSA & $\boldsymbol{g}_{u,\ell}$ in TBCG) perform worse than NextIP(w/o BSA & $\boldsymbol{x}^*_{u,\ell}$ in TBCG), the reason might be that when

Architecture	UB	Tmall
NextIP(w/o TBCG&BSA)	0.556	0.616
NextIP(w/o TBCG)	0.577	0.635
NextIP(w/o BSA)	0.624	0.678
NextIP(w/o BSA & $g_{u,\ell}$ in TBCG)	0.557	0.634
NextIP(w/o BSA & $x^*_{u,\ell}$ in TBCG)	0.570	0.648
NextIP	**0.632**	**0.681**

Table 5: Recommendation performance (Rec@10) of our NextIP with different architectures on UB and Tmall for ablation studies.



(a) UB



(b) Tmall

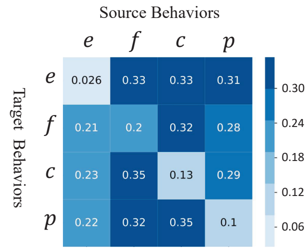
Figure 3: Recommendation performance (Rec@10) of our NextIP and its variants by removing different behavior-specific item sub-sequences on Tmall and UB.

only using one behavior-specific context information in predicting the corresponding behaviors, the behavior-specific item sequence encoders are hard to get trained well.
- Our NextIP performs the best compared with all of its variants, which clearly demonstrates the positive complementary effect of all the designed components in our NextIP.

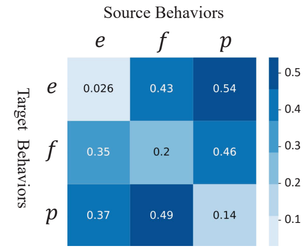
5.3.3 Impact of Behavior-Specific Item Sequences (RQ3). In order to study the impact of different behavior-specific item sequences on the performance of our NextIP, we conduct an additional ablation study and report the results in Figure 3. By removing each kind of behavior-specific item sequences in task 1, we can get four variants of our NextIP, i.e., "-e", "-f", "-c" and "-p". We have the following observations.

- The performance decline of removing the examination-specific item sequences (i.e., "-e") is more significant than that of "-f" or "-c", because the examinations are much more abundant.
- Our NextIP performs the best compared with all of its variants, which again clearly demonstrates the effectiveness and rationality of utilizing all the behavior-specific item sequences in our NextIP.

5.3.4 Case Study (RQ4). To investigate the adaptive knowledge transfer of the context information in different behavior-specific item sequences in predicting the next item of different types of target behaviors, we visualize the attention scores of different source behaviors to different target behaviors in TBCG. Specifically, after the model converges, we obtain the attention scores of all the users used for predicting different target behaviors in the training data. Finally, we visualize the mean attention scores from all the users in each data as a heatmap matrix shown in Figure 4. Note that the horizontal axis represents the source behaviors and the vertical



(a) **UB**



(b) **Tmall**

Figure 4: Visualization of the attention scores of different source behaviors to different target behaviors in target-behavior aware context aggregator (TBCG) of our NextIP on UB and Tmall.

axis represents the target behaviors. We can have the following observations.

- The main diagonal elements of both heatmap matrices have the lowest attention scores in corresponding rows, which is reasonable since $x^*_{u,\ell}$ is added to the final context representation as shown in Eq.(9).
- When predicting purchases, the information contribution of adds-to-favorite and adds-to-cart is larger than that of examinations on both datasets, which is consistent with our intuition in online shopping.
- When predicting adds-to-cart, the behaviors of adds-to-favorite are more informative than purchases in providing knowledge, because the interests reflected in the purchase-specific item sequences may have expired to some extent. In general, these observations further demonstrate the rationality and interpretability of our NextIP.

6 CONCLUSIONS AND FUTURE WORK

In this work, we propose a novel framework named NextIP for multi-behavior sequential recommendation, which adopts a dual-task learning strategy to convert the problem to a next-item prediction task and a purchase prediction task. Specifically, we design a target-behavior aware context aggregator (TBCG) to transfer the unique knowledge of different behaviors so as to predict the next interaction in a behavior-aware manner more accurately. Moreover, we design a behavior-aware self-attention (BSA) mechanism to aggregate the collection of historical interacted items of each behavior and treat them as negative samples for more accurate learning of purchase-oriented preferences.

For future works, we are interested in extending our NextIP to incorporate knowledge graphs about users and items.

ACKNOWLEDGMENTS

We thank the support of National Natural Science Foundation of China Nos. 62172283 and 61836005. We thank Miss Jing Lin for her data preprocessing scripts and Mr. Dugang Liu for his helpful discussions.

REFERENCES

[1] Jianxin Chang, Chen Gao, Yu Zheng, Yiqun Hui, Yanan Niu, Yang Song, Depeng Jin, and Yong Li. 2021. Sequential Recommendation with Graph Neural Networks. In *Proceedings of the 44th International ACM SIGIR Conf. on Research and Development in Information Retrieval*. 378–387.

[2] Chong Chen, Weizhi Ma, Min Zhang, Zhaowei Wang, Xiuqiang He, Chenyang Wang, Yiqun Liu, and Shaoping Ma. 2021. Graph Heterogeneous Multi-Relational Recommendation. In *Proceedings of the 35th AAAI Conf. on Artificial Intelligence*. 3958–3966.

[3] Chong Chen, Min Zhang, Yongfeng Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2020. Efficient Heterogeneous Collaborative Filtering without Negative Sampling for Recommendation. In *Proceedings of the 34th AAAI Conf. on Artificial Intelligence*. 19–26.

[4] Xiancong Chen, Lin Li, Weike Pan, and Zhong Ming. 2020. A Survey on Heterogeneous One-class Collaborative Filtering. *ACM Transactions on Information Systems* 38, 4 (2020), 35:1–35:54.

[5] Tim Donkers, Benedikt Loepp, and Jürgen Ziegler. 2017. Sequential User-based Recurrent Neural Network Recommendations. In *Proceedings of the 11t ACM Conf. on Recommender Systems*. 152–160.

[6] Chen Gao, Xiangnan He, Dahua Gan, Xiangning Chen, Fuli Feng, Yong Li, Tat-Seng Chua, and Depeng Jin. 2019. Neural Multi-Task Recommendation from Multi-Behavior Data. In *Proceedings of the 35th IEEE International Conference on Data Engineering*. 1554–1557.

[7] Yulong Gu, Zhuoye Ding, Shuaiqiang Wang, Lixin Zou, Yiding Liu, and Dawei Yin. 2020. Deep Multifaceted Transformers for Multi-Objective Ranking in Large-Scale E-Commerce Recommender Systems. In *Proceedings of the 29th ACM International Conf. on Information and Knowledge Management*. 2493–2500.

[8] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition*. 9726–9735.

[9] Ruining He, Wang-Cheng Kang, and Julian McAuley. 2017. Translation-based Recommendation. In *Proceedings of the 11t ACM Conf. on Recommender Systems*. 161–169.

[10] Ruining He and Julian McAuley. 2016. Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation. In *Proceedings of the 16th IEEE International Conf. on Data Mining*. 191–200.

[11] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conf. on World Wide Web*. 173–182.

[12] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. In *Proceedings of the 27th ACM International Conf. on Information and Knowledge Management*. 843–852.

[13] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *Proceedings of the 4th International Conf. on Learning Representations*.

[14] Bowen Jin, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. 2020. Multi-Behavior Recommendation with Graph Convolutional Networks. In *Proceedings of the 43rd International ACM SIGIR Conf. on Research and Development in Information Retrieval*. 659–668.

[15] Santosh Kabbur, Xia Ning, and George Karypis. 2013. FISM: Factored Item Similarity Models for Top-N Recommender Systems. In *Proceedings of the 19th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining*. 659–667.

[16] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *Proceedings of the 18th IEEE International Conf. on Data Mining*. 197–206.

[17] Hui Li, Ye Liu, Nikos Mamoulis, and David S. Rosenblum. 2020. Translation-Based Sequential Recommendation for Complex Users on Sparse Data. *IEEE Transactions on Knowledge and Data Engineering* 32, 8 (2020), 1639–1651.

[18] Zhi Li, Hongke Zhao, Qi Liu, Zhenya Huang, Tao Mei, and Enhong Chen. 2018. Learning from History and Present: Next-item Recommendation via Discriminatively Exploiting User Behaviors. In *Proceedings of the 24th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining*. 1734–1743.

[19] Jing Lin, Weike Pan, and Zhong Ming. 2020. FISSA: Fusing Item Similarity Models with Self-Attention Networks for Sequential Recommendation. In *Proceedings of the 14th ACM Conf. on Recommender Systems*. 130–139.

[20] Babak Loni, Roberto Pagano, Martha Larson, and Alan Hanjalic. 2016. Bayesian Personalized Ranking with Multi-Channel User Feedback. In *Proceedings of the 10th ACM Conference on Recommender Systems*. 361–364.

[21] Wanqi Ma, Xiancong Chen, Weike Pan, and Zhong Ming. 2022. VAE++: Variational AutoEncoder for Heterogeneous One-Class Collaborative Filtering. In *Proceedings of the 15th International Conference on Web Search and Data Mining*. 666–674.

[22] Wenjing Meng, Deqing Yang, and Yanghua Xiao. 2020. Incorporating User Micro-behaviors and Item Knowledge into Multi-task Learning for Session-based Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conf. on Research and Development in Information Retrieval*. 1091–1100.

[23] Weike Pan, Hao Zhong, Congfu Xu, and Zhong Ming. 2015. Adaptive Bayesian Personalized Ranking for Heterogeneous Implicit Feedbacks. *Knowledge-Based Systems* 73 (2015), 173–180.

[24] Ruihong Qiu, Jingjing Li, Zi Huang, and Hongzhi YIn. 2019. Rethinking the Item Order in Session-based Recommendation with Graph Neural Networks. In *Proceedings of the 28th ACM International Conf. on Information and Knowledge Management*. ACM, 579–588.

[25] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking From Implicit Feedback. In *Proceedings of the 25th Conf. on Uncertainty in Artificial Intelligence*. 452–461.

[26] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing Personalized Markov Chains for Next-basket Recommendation. In *Proceedings of the 19th International Conf. on World Wide Web*. 811–820.

[27] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *Proceedings of the 28th ACM International Conf. on Information and Knowledge Management*. 1441–1450.

[28] Jiaxi Tang and Ke Wang. 2018. Personalized top-N Sequential Recommendation via Convolutional Sequence Embedding. In *Proceedings of the 11th ACM International Conf. on Web Search and Data Mining*. 565–573.

[29] Md. Mehrab Tanjim, Congzhe Su, Ethan Benjamin, Diane Hu, Liangjie Hong, and Julian J. McAuley. 2020. Attentive Sequential Models of Latent Intent for Next Item Recommendation. In *Proceedings of the 29th International Conf. on World Wide Web Companion*. 2528–2534.

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Annual Conf. on Neural Information Processing Systems*. 5998–6008.

[31] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian. 2022. A Survey on Session-based Recommender Systems. *ACM Computing Surveys (CSUR)* 54, 7 (2022), 154:1–154:38.

[32] Wen Wang, Wei Zhang, Shukai Liu, Qi Liu, Bo Zhang, Leyu Lin, and Hongyuan Zha. 2020. Beyond Clicks: Modeling Multi-Relational Item Graph for Session-Based Target Behavior Prediction. In *Proceedings of the 29th International Conf. on World Wide Web Companion*. 3056–3062.

[33] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. 2020. Global Context Enhanced Graph Neural Networks for Session-based Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conf. on research and development in Information Retrieval*. 169–178.

[34] Wei Wei, Chao Huang, Lianghao Xia, Yong Xu, Jiashu Zhao, and Dawei Yin. 2022. Contrastive Meta Learning with Behavior Multiplicity for Recommendation. In *Proceedings of the 15th ACM WSDM International Conference on Web Search and Data Mining*. 1120–1128.

[35] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based Recommendation with Graph Neural Networks. In *Proceedings of the 33th AAAI Conf. on Artificial Intelligence*. 346–353.

[36] Yiqing Wu, Ruobing Xie, Yongchun Zhu, Xiang Ao, Xin Chen, Xu Zhang, Fuzhen Zhuang, Leyu Lin, and Qing He. 2022. Multi-view Multi-behavior Contrastive Learning in Recommendation. In *Database Systems for Advanced Applications - 27th International Conference*, Vol. 13246. 166–182.

[37] Lianghao Xia, Chao Huang, Yong Xu, Peng Dai, Xiyue Zhang, Hongsheng Yang, Jian Pei, and Liefeng Bo. 2021. Knowledge-Enhanced Hierarchical Graph Transformer Network for Multi-Behavior Recommendation. In *Proceedings of the 35th AAAI Conf. on Artificial Intelligence*. 4486–4493.

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

[39] Ruobing Xie, Cheng Ling, Yalong Wang, Rui Wang, Feng Xia, and Leyu Lin. 2020. Deep Feedback Network for Recommendation. In *Proceedings of the 29th International Joint Conf. on Artificial Intelligence*. 2519–2525.

[40] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. 2019. Graph Contextualized Self-attention Network for Session-based Recommendation. In *Proceedings of the 28th International Joint Conf. on Artificial Intelligence*. 3940–3946.

[41] Haoran Yang, Hongxu Chen, Lin Li, Philip S. Yu, and Guandong Xu. 2021. Hyper Meta-Path Contrastive Learning for Multi-Behavior Recommendation. In *Proceedings of the 37th IEEE International Conference on Data Engineering*. IEEE, 787–796.

[42] Bo Yu, Ruoqian Zhang, Wei Chen, and Junhua Fang. 2022. Graph Neural Network based Model for Multi-Behavior Session-based Recommendation. *GeoInformatica* 26, 2, 429–447.

[43] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the 12th ACM International Conf. on Web Search and Data Mining*. 582–590.

[44] Meizi Zhou, Zhuoye Ding, Jiliang Tang, and Dawei Yin. 2018. Micro Behaviors: A New Perspective in E-commerce Recommender Systems. In *Proceedings of the 11th ACM International Conf. on Web Search and Data Mining*. 727–735.