Modelling Temporal Dynamics and Repeated Behaviors for Recommendation

Xin Zhou[1], Zhu Sun[2], Guibing Guo[1(✉)], and Yuan Liu[1]

[1] Department of Software Engineering, Northeastern University, Shenyang, China
xinzhou@stumail.neu.edu.cn, {guogb,liuyuan}@swc.neu.edu.cn
[2] Department of Computing, Macquarie University, Sydney, Australia
z.sun@mq.edu.au

Abstract. Personalized recommendation has yield immense success in predicting user preference with heterogeneous implicit feedback (HIF), i.e., various user behaviors. However, existing studies consider less about the temporal dynamics and repeated patterns of HIF. They simply suppose: (1) a hard rule among user behaviors (e.g., *add-to-cart* must come before *purchase* and after *view*); (2) merge repeated behaviors into one (e.g., *view* several times is considered as *view* once only), thus failing to unveil user preferences from their real behaviors. To ease these issues, we, therefore, propose a novel end-to-end neural framework – TDRB, which automatically models the *Temporal Dynamics and Repeated Behaviors* to assist in capturing user preference, thus achieving more accurate recommendations. Empirical studies on three real-world datasets demonstrate the superiority of our proposed TDRB against other state-of-the-arts.

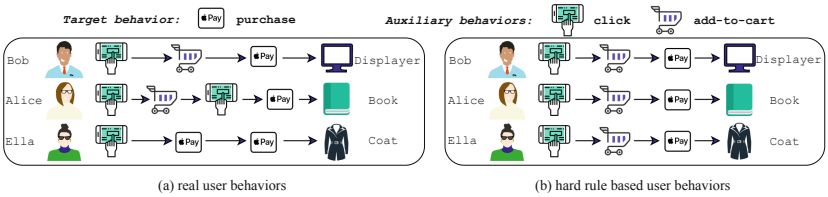
Keywords: Temporal dynamics · Repeated behaviors · Heterogeneous implicit feedback · Recommendation

1 Introduction

Recommender systems have recently become prominent tools to provide personalized services for customers, so as to alleviate the *information overload* problem [3]. A number of recommenders [2,3,10,11] have been proposed to help infer users' potential interests based on their heterogeneous implicit feedback (HIF). Taking e-commerce as an example, the recommenders mainly leverage users' historical behaviors of various types (e.g., *view, click, add-to-cart, purchase*) to help predict what product to *purchase* afterwards. In this scenario, *purchase* is the *target behavior*, which directly reflects users' preference towards products. In contrast, *view, click, add-to-cart* are *auxiliary behaviors* to indirectly suggest users' taste over products to some extent. Both target and auxiliary behaviors benefit for user preference inference, possibly with different degrees.

X. Zhou and Z. Sun—Contribute equally and share the co-first authorship.

© Springer Nature Switzerland AG 2020
H. W. Lauw et al. (Eds.): PAKDD 2020, LNAI 12084, pp. 181–193, 2020.
https://doi.org/10.1007/978-3-030-47426-3_15



(a) real user behaviors (b) hard rule based user behaviors

Fig. 1. Runing examples for sequential user behaviors on products in e-commerce, where (a) depicts the real sequential behaviors towards products; (b) illustrates the hard rule based sequential behaviors on products.

Running Example. To illustrate, Fig. 1(a) depicts real user behaviors on products in an e-commerce system (e.g. Amazon), where two essential characteristics are noted: (1) `Temporal dynamics`: users perform sequential behaviors over products with different orders, indicating different behavior patterns. For instance, Bob and Alice prefer *add-to-cart* before *purchase*, whilst Ella directly buys the 'Coat' after *click* without *add-to-cart*. (2) `Repeated behaviors:` users may perform certain behaviors several times over products, reflecting a reinforced preference to some degree. For example, Alice clicks twice to check details before she makes the purchase decision; Ella may be quite satisfied with the quality after purchasing the 'Coat' and directly buy another one for her friend.

The above examples highlight the presence of temporal dynamics and repeated behaviors in user-item interactions, which could potentially facilitate to model user preference, thus achieving better recommendations. They, however, are not well investigated by existing HIF based studies: (1) most methods [1,5,11–14] directly ignore the temporal dynamics. They only model the influence of limited types of user behaviors (e.g., only *view* and *purchase*) via a weighted combination, which heavily restricts the generability of these methods; (2) although several approaches [2,10] notice the temporal dynamics, they simply pre-define a hard rule to order various user behaviors, thus failing to capture the real temporal dynamics. For instance, they assume that *add-to-cart* must come before *purchase* and after *click*. In Fig. 1(b), all users share the same behavior pattern based on the hard rule: *click* → *add-to-cart* → *purchase* regardless of what their real behaviors are; and (3) none of them considers the repeated user behaviors. They merely merge the duplicated user-item interactions by keeping the earliest one. In Fig. 1, *click* twice is only treated as *click* once for Alice. In this sense, there's much room left to better exploit the temporal dynamics and repeated behaviors in HIF based recommendation.

In this paper, we, therefore, propose a novel end-to-end neural framework – TDRB, which exploits both *Temporal Dynamics and Repeated Behaviors* to truly uncover user behavior patterns, thus capturing user preference over items in a more accurate manner. Specifically, TDRB is composed of three modules, (1) `Target Module` employs outer product and a convolutional neural network

to directly learn expressive and high-order user-item correlation (i.e., user preference over items) regarding to the target behavior (e.g., *purchase*); (2) `Auxilary Module` aims to assist target module in predicting user preference over items. It uses gated recurrent units to accurately model the temporal dynamics and repeated patterns of auxiliary behaviors that happen before its corresponding target behavior; and (3) `Fusion Module` devises three strategies (i.e., MLP [6], Attention mechanism [9] and Outer product) to seamlessly integrate both target and auxiliary behaviors for a more accurate user preference estimation.

To conclude, our main contributions lie in three folds: (1) to the best of our knowledge, we are the first to exploit temporal dynamics and repeated behaviors in HIF based recommendation; (2) we propose a novel end-to-end neural framework – TDRB, which delicately models the temporal dynamics and repeated behaviors for user preference prediction, thus achieving high-quality recommendation; and (3) extensive experiments on three real-world datasets show the superiority of our proposed TDRB, which significantly beats state-of-the-arts with a lift of 27.91%, 61.32% w.r.t. HR and NDCG on average, respectively.

2 Related Work

The heterogenous implicit feedback (HIF)-based recommenders can be broadly classified into conventional recommenders and deep learning-based recommenders.

Conventional Recommenders. Early works solely leverage target behavior for the recommendation. For example, Rendle et al. proposed BPRMF [15] to maximize the difference of user preference over items with and without target behavior. After that, many variants have been proposed based upon BPRMF, such as GBPR [11], ABPR [12] and eALS [5]. However, they all fail to fully exploit the HIF by taking into account target behaviors only. Later, some researchers attempted to employ auxiliary behaviors (e.g., *view, click*) in addition to target behavior for the performance-enhanced recommendation. For instance, Qiu et al. designed TBPR/BPRH [13,14] to utilize *purchase, view* and *like* for trinity preference ranking. Ding et al. proposed VALS [1] by fusing *purchase* with *view* through manually pre-defined weights. GcBPR [3] resolved the data sparsity problem by generating target behavior from a linear regression of auxiliary behaviors. Yin et al. devised SPTF [18] to overcome the issue of heavy skewness of the interaction distribution w.r.t. different types of HIF. Nevertheless, these methods all ignore the temporal dynamics and repeated patterns of user behaviors for the recommendation. Furthermore, most of them only consider limited types of auxiliary behaviors, thus restricting their generability. Afterwards, Loni et al. proposed McBPR [10] to capture the temporal dynamics by a pre-defined hard rule (e.g., *add-to-cart* must occur before *purchase* and after *click*), as illustrated in Fig. 1(b). Undoubtedly, the proposed hard rule cannot truly express the temporal dynamics and repeated user behaviors.

Deep Learning-Based Recommenders. Deep learning has made great breakthroughs in various related areas, such as image recognition [4] and natural

language processing [19]. Being capable of capturing the non-linearity of user-item interactions, they have been widely applied to the recommendation with HIF [6,7]. Similar to conventional recommenders, early deep learning-based recommenders only model user preference with target behavior, such as NCF [6]. Soon afterwards, He et al. further designed ConvNCF [7] to use an outer product and convolutional neural network (CNN) to learn high-order correlations among user-item interactions. Later, some studies turn to fuse both target and auxiliary behaviors. Wen et al. [17] devised a neural framework to capture both linearity and non-linearity of heterogeneous behaviors through. Recently, Gao et al. developed a neural model named NTMR [2] encoding a hard rule-based order for heterogeneous user behaviors, that is, *add-to-cart* must appear before *purchase* and after *view*. All of these methods mentioned above, however, fail to model the temporal dynamics and repeated behavior for a high-quality recommendation.

Note that, there are also some works model the temporal dynamics of item sequences, instead of user behavior sequences. For instance, GRU4rec [8], NARM [9] and SLRC [16] model the temporal dynamic of repeated purchased items for next item recommendation, which is out of our scope, i.e., exploiting temporal dynamics of repeated behaviors for general item recommendation.

3 The Proposed TDRB

This section presents the proposed neural framework – TDRB for heterogeneous implicit feedback (HIF)-based recommendation. It fully exploits the temporal dynamics and repeated behaviors to capture user preference in a more accurate fashion, thus achieving high-quality recommendation.

3.1 Problem Formulation

Notations. Let u, i, f, a denote user u, item i, target behavior f and auxiliary behavior a, respectively; $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{O}$ are user, item and auxiliary behavior embedding matrices; $\mathbf{p}_u$, $\mathbf{q}_i$, $\mathbf{o}_a$ denote the corresponding embedding vectors for u, i, a. Throughout this paper, we use bold uppercase letter to denote a matrix (e.g., $\mathbf{P}$) and bold lowercase letter (e.g., $\mathbf{p}_u$) to denote a vector.

Data Segmentation. Most existing studies [1,6,7] consider less about the temporal dynamics of user behaviors, and always merge the repeated (both target and auxiliary) behaviors by remaining the earliest one. However, we contend that the temporal dynamics are capable of reflecting user behavior patterns, and the repeated behaviors may indicate a reinforced user preference. Hence, we conduct data segmentation to simulate the real scenario better. Given all behaviors between (u, i) pair ordered by time, for example, $B(u, i) = \{a_1, a_2, a_3, f, a_4, f, f\}$, we first split them into m sequences by target behavior, e.g., $S_1(u, i) = \{a_1, a_2, a_3, f\}$, $S_2(u, i) = \{a_4, f\}$ and $S_3(u, i) = \{f\}$, such that each sequence consists of one target behavior and all the auxiliary behaviors before it if available. m is the number of target behaviors in $B(u, i)$. We then feed all the segmented sequences, e.g., $S_1(u, i), S_2(u, i)$ and $S_3(u, i)$,

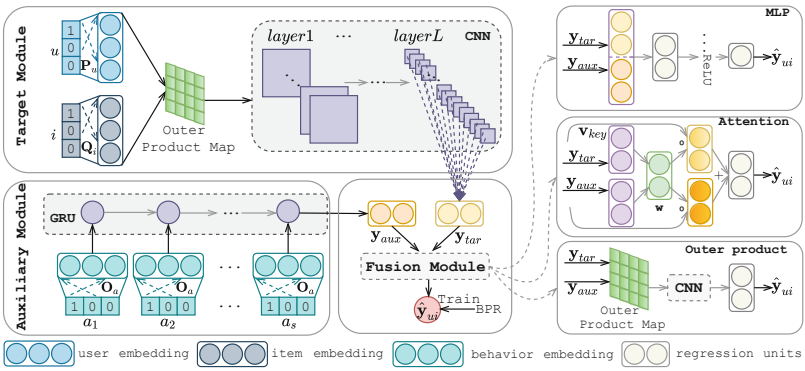


Fig. 2. The overall framework of our proposed TDRB, which is composed of three modules: (1) target module in the upper left corner; (2) auxilary module in the bottom left corner; and (3) fusion module in the bottom center. The three different fusion strategies (i.e., MLP, Attention and Outer product) are depicted on the right side.

between (u, i) pair for model training, to estimate the preference score of u over i, i.e., $\hat{y}_{ui}$, and generate a personalized ranking list with top-N items based on the preference scores. By doing so, the temporal dynamics and repeated (both target and auxiliary) behaviors can be preserved to the utmost.

3.2 Different Modules of TDRB

The overall framework of TDRB is illustrated by Fig. 2, mainly composed of three modules, namely (1) `Target Module` directly captures the high-order user-item correlation from the perspective of target behavior through the outer product and convolutional neural network (CNN) [7]; (2) `Auxiliary Module` utilizes gated recurrent units to encode temporal dynamics and repeated patterns of auxiliary behaviors, so as to enhance target module; and (3) `Fusion Module` unifies both target and auxiliary behaviors via three different strategies (i.e., MLP, Attention and Outer product) for high-quality recommendation. As mentioned in Data Segmentation, there are m segmented sequences for a (u, i) pair. For ease of presentation, we take one segmented sequence $S(u, i) = \{a_1, a_2, \ldots, a_s, f\}$ as an example to demonstrate different modules of TDRB. Note that $S(u, i)$ has and only has one target behavior, but may contain multiple auxiliary behaviors as well as repeated auxiliary behaviors.

Target Module. It aims to directly model u's preference over i regarding the target behavior f. We first project u and i into dense embedding space by,

$$\mathbf{p}_u = \mathbf{P}^T \mathbf{v}_u, \mathbf{q}_i = \mathbf{Q}^T \mathbf{v}_i \tag{1}$$

where $\mathbf{p_u}, \mathbf{q_i} \in \mathbb{R}^{1\times 32}$ are dense embeddings for u, i; $\mathbf{v}_u, \mathbf{v}_i$ are one-hot sparse vectors for u, i. Different from existing neural recommenders [5,6] combining user

and item embeddings via a simple concatenation or element-wise product, we are inspired by [7] to utilize outer product above the user and item embedding layers. It results in a 2D interaction map, which is more expressive and capable of capturing the high-order user-item correlation, given by,

$$\psi(\mathbf{p}_u, \mathbf{q}_i) = \mathbf{p}_u \otimes \mathbf{q}_i \tag{2}$$

where $\otimes$ denotes outer product operation; $\psi(\mathbf{p}_u, \mathbf{q}_i) \in \mathbb{R}^{32\times32}$ denotes the interaction map. Thanks to the superior capability of extracting local relations of graph features, we then apply a CNN on the interaction map, so as to fully capture the latent features of $\psi(\mathbf{p}_u, \mathbf{q}_i)$.

The structure of CNN is fine-tuned based on literature [7]. To be more specific, it has 5 hidden layers and 32 feature maps for each hidden layer. We set the kernel size as $2*2$ and the stride to be 2. Hence, the feature dimension is halved layer by layer. The input for the first layer is a 2D matrix, i.e., $\psi(\mathbf{p}_u, \mathbf{q}_i) \in \mathbb{R}^{32\times32}$, and its output is a 3D tensor. For the rest layers, both input and output are 3D tensors. The element $e^l_{x,y,k}$ of the x^{th} row and y^{th} column of feature map in the k^{th} filter $\mathbf{f}$ for layer l is defined as,

$$e^l_{x,y,k} = \varphi(\mathbf{b}_l + \sum\nolimits_{a=0}^{1}\sum\nolimits_{b=0}^{1} e_{2x+a,2y+b} \cdot \mathbf{f}^l_{1-a,1-b,k}) \tag{3}$$

where $\mathbf{f}^l_{1-a,1-b,k} \in \mathbb{R}^{2\times2\times32}$ if $l = 1$; otherwise $\mathbf{f}^l_{1-a,1-b,k} \in \mathbb{R}^{2\times2\times32\times32}$ when $1 < l \leq 5$; $\mathbf{b}_l$ denotes bias; k is the number of kernels; $\varphi(x)$ denotes the activation function (ReLU [8]). The final output of target module can be defined by Eq. (4), where g_θ denotes CNN with parameters $\theta = \{\mathbf{b}, \mathbf{f}\}$.

$$\mathbf{y}_{tar} = g_\theta(\psi(\mathbf{p}_u, \mathbf{q}_i)) \tag{4}$$

Auxiliary Module. Auxiliary behaviors (i.e., *view, click*) could indirectly reflect user inclination, thus benefiting for a better recommendation. Hence, the auxiliary module mainly takes advantage of auxiliary behaviors for a more fine-grained user preference inference. Existing studies either ignore the temporal dynamics of auxiliary behaviors [1,3,13,14], or pre-define a hard rule to order them [2,10]. Besides, all of them overlook the repeated behaviors. To ease this issue, we consider to accommodate the real auxiliary behavior sequence $S_a(u,i) = \{a_1, a_2, \ldots, a_s\}$, which can be obtained by removing the target behavior f from the segmented sequence $S(u,i)$. The gated recurrent units (GRU) [8] are applied to help model the temporal dynamics and repeated patterns encoded in the auxiliary behavior sequence $S_a(u,i)$, as GRU has proven to be more adaptive and stable with fewer parameters in comparison with recurrent neural network (RNN) and long short term memory (LSTM) [8].

Given $S_a(u,i) = \{a_1, a_2, \ldots, a_s\}$ as input, we project each auxiliary action a_t in the sequence into dense embedding space, given by,

$$\mathbf{o}_{a_t} = \mathbf{O}^T \mathbf{v}_{a_t} \tag{5}$$

where $\mathbf{o}_{a_t} \in \mathbb{R}^{1\times32}$ is the embedding vector of a_t; $\mathbf{v}_{a_t}$ is the one-hot vector for a_t. The embedding of each auxiliary behavior is then considered as the input of

GRU at each time step. In this case, if any duplicated auxiliary behaviors exist in $S_a(u,i)$, their embeddings will be fed into GRU repeatedly, so as to reinforce their influence automatically. Specifically, at time state t, the hidden state $\mathbf{h}_t$ of GRU is updated as follows,

$$\mathbf{h}_t = (1-z) \circ \mathbf{h}_{t-1} + z \circ \tau(\mathbf{o}_{at}\mathbf{U}_h + (\mathbf{h}_{t-1} \circ r)\mathbf{W}_h + b_h) \tag{6}$$

where $\tau(x)$ is the tanh activation function; $\circ$ is the multiplication between a vector and a scalar; $z = \sigma(\mathbf{o}_{at}\mathbf{U}_z + \mathbf{h}_{t-1}\mathbf{W}_z + b_z)$ and $r = \sigma(\mathbf{o}_{at}\mathbf{U}_r + \mathbf{h}_{t-1}\mathbf{W}_r + b_r)$ are the update gate and reset gate of GRU, respectively; $\delta = \{\mathbf{U}_z, \mathbf{W}_z, b_z, b_r, b_h\}$ is the parameter set of GRU. The last hidden state at time step s is supposed to be the output $\mathbf{y}_{aux}$, i.e., $\mathbf{y}_{aux} = \mathbf{h}_s$, which represents the influence of auxiliary behaviors on user preference. As the input sequence $S_a(u,i)$ is ordered by time and may contain repeated behaviors, the auxiliary module is capable of better capture temporal dynamics and repeated patterns of user behaviors. Note that, if there is only target behavior and no auxiliary behaviors between u and i, i.e., $S(u,i) = \{f\}$ and $S_a(u,i) = \emptyset$, we set $\mathbf{o}_{a_t}$ as an all-zero vector.

Fusion Module. It aims to automatically incorporate the influence of both target (i.e., $\mathbf{y}_{tar}$) and auxiliary (i.e., $\mathbf{y}_{aux}$) behaviors for a more comprehensive modeling on u's preference towards i, i.e., $\hat{\mathbf{y}}_{ui}$. Three different fusion strategies are thus devised for a better exploration as introduced below.

(1) `MLP`: a straightforward way is to adopt MLP on the concatenated $\mathbf{y}_{tar}$ and $\mathbf{y}_{aux}$, as shown in the upper right corner of Fig. 2. Following [6], we implement MLP with a tower structure, halving the layer size for each successive higher layer. Hence the estimated preference score $\hat{\mathbf{y}}_{ui}$ is,

$$\hat{\mathbf{y}}_{ui} = \zeta_{out}(\zeta_l(...\zeta_2(\zeta_1(\psi(\mathbf{y}_{tar}, \mathbf{y}_{aux})))...)) \tag{7}$$

where ζ_{out} and ζ_l denote the output layer and the l^{th} layer in MLP.

(2) `Attention`: it fuses $\mathbf{y}_{tar}$ and $\mathbf{y}_{aux}$ by distinguishing their weights in an automatic fashion, as shown in the center right of Fig. 2. The attention weights $\mathbf{w} \in \mathbb{R}^{1\times 2}$ and $\hat{\mathbf{y}}_{ui}$ are calculated by Eq. (8), where $\xi(x)$ is the softmax function; $\phi_\vartheta(x)$ is the regression function.

$$\begin{aligned} \mathbf{w} &= [w_1, w_2] = \xi(\textstyle\sum_x^k \mathbf{v}_{key}^x \cdot [\mathbf{y}_{aux}^x, \mathbf{y}_{tar}^x]), \\ \hat{\mathbf{y}}_{ui} &= \phi_\vartheta(w_1 \circ \mathbf{y}_{tar} + w_2 \circ \mathbf{y}_{tar}) \end{aligned} \tag{8}$$

(3) `Outer product`: similar as described in the target module, it utilizes outer product with CNN to capture and extract more expressive and high-order correlation between $\mathbf{y}_{tar}$ and $\mathbf{y}_{aux}$, shown in the bottom right corner of Fig. 2. The user preference score $\hat{\mathbf{y}}_{ui}$ is re-defined as follows,

$$\hat{\mathbf{y}}_{ui} = \phi_\vartheta(g_{\theta_1}(\psi(\mathbf{y}_{tar}, \mathbf{y}_{aux}))) \tag{9}$$

Optimization. Following state-of-the-arts [3,7,15], we formulate top-N recommendation as a pair-wise ranking problem, where user u prefers positive

item i (with target behavior) to negative item j (without target behavior). Hence, we sample triplet $\mathcal{D}_s = \{(u, i, j)\}$ for training and keep the sample ratio between items i, j as $1:1$. Finally, we minimize the objective function shown in Eq. (10), where Θ is the set of model parameters to update.

$$L = \sum\nolimits_{(u,i,j)\in\mathcal{D}_s} -\ln\sigma(\widehat{y}_{ui} - \widehat{y}_{uj}) + \lambda_\Theta ||\Theta||^2 \tag{10}$$

4 Experiments and Analysis

We conduct extensive experiments on three real-world datasets with the goal of answering four research questions: (RQ1) Does modeling temporal dynamics and repeated behaviors enhance the performance of TDRB? (RQ2) How do different fusion strategies affect the performance of TDRB? (RQ3) Does the proposed TDRB outperform state-of-the-art algorithms?

4.1 Experiment Setup

Datasets. Three real-world datasets are used for evaluation: `Xing`[1], `Taobao2014`[2] and `Taobao2017`[3]. In particular, Xing records the data from a job-hunting website and Taobao2014/2017 are obtained from Taobao (www.taobao.com.). They are different in three aspects: (1) Xing is much denser than

Table 1. Statistics of the two utilized datasets

Datasets	#Users	#Items	#Records	Density	#Behaviors	Behavior types
Xing	459	5,932	12,343	0.453%	4	click, bookmark, remove, reply
Taobao2014	1,569	811,063	1,052,488	0.082%	4	click, collect, add-to-cart, payment
Taobao2017	17,793	309,250	803,939	0.014%	4	click, favorite, add-to-cart, buy

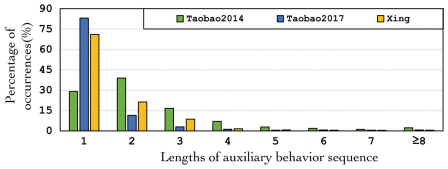


Fig. 3. Length distribution of the auxiliary behavior sequence.

[1] https://github.com/recsyschallenge/2016/.
[2] https://tianchi.aliyun.com/dataset/dataDetail?dataId=46.
[3] https://tianchi.aliyun.com/dataset/dataDetail?dataId=649.

Taobao2014/2017, as shown in Table 1; (2) in regard to the behavior types, in Xing *reply* is the target behavior; *click*, *bookmark* and *remove* are the auxiliary behaviors; and in Taobao2014, *payment* is the target behavior; *click*, *collect* and *add-to-cart* are the auxiliary behaviors; whilst in Taobao2017, the target behavior is *buy* and the auxiliary behaviors include *click, favorite, add-to-cart*; and (3) the distributions among the three datasets w.r.t. the length of auxiliary behavior sequences (i.e., $S_a(u, i)$) is quite different, as depicted in Fig. 3. The length of most sequences (above 90%) ranges from 1 to 4 on Taobao2014, whilst in Xing and Taobao2017, most sequences (90% or so) are shorter than 3. To balance the quantity and quality of each dataset, we filter out users with less than 3 target behaviors and items with less than 6, 2, 10 target behaviors for Xing, Taobao2014, and Taobao2017, respectively. The statistics of the processed datasets are summarized in Table 1.

Evaluation Protocols. We adopt the widely-used leave-one-out for evaluation [1,5–7]. For each user, we hold out her latest interaction as the test set; the second latest as validate set; and utilize the remaining data as the train set. In order to improve the test efficiency, we follow the common strategy [6,7], by randomly sampling 999 negative items that user has not performed target behavior on, and rank the test item among the 1000 items. HR@N and NDCG@N [6] are adopted to evaluate the ranking performance, where $N = \{5, 10, 20\}$.

Comparison Methods. We compare TDRB with the following state-of-the-arts, including (1) **MostPopular** is a non-personalized method that recommends the most popular items w.r.t. target behaviors to users; (2) **BPRMF** [15] is a pair-wise learning algorithm based on MF, which only considers the influence of target behavior on user preference prediction; (3) **McBPR** [10] is the multi-channel based BPR, which divides heterogeneous behaviors into different channels; (4) **ConvNCF** [7] is a deep learning-based recommender adopting outer product and CNN to learn high-order correlations among embedding dimensions; (5) **NMTR** [2] is a recently proposed deep learning-based recommender that utilizes a pre-defined hard rule to order different types of behaviors, that is, *add-to-cart*/*bookmark* shows before *purchase*/*reply* and after *click*.

Parameter Settings. The optimal parameter settings for all the comparison methods are achieved by the empirical study or suggested by the original papers. For a fair comparison, we set the embedding size and hidden state size to 32; a grid search in $[10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2]$ is applied to find out the best settings

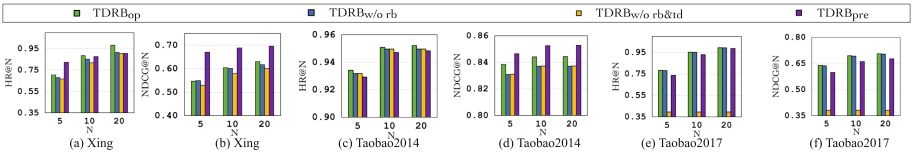


Fig. 4. Impacts of modeling temporal dynamics and repeated behaviors on TDRB.

for the regularization coefficient to avoid over-fitting; the learning rate for updating all embeddings (i.e., $\mathbf{P}, \mathbf{Q}, \mathbf{O}$) is tuned in the range of $[5^{-4}, 5^{-3}, 5^{-2}]$; and for other neural network parameters, it is tuned within $[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]$. For all methods besides MostPop and BPRMF, we pretrain the user and item embeddings via BPRMF. For our proposed TDRB, it is implemented with Tensorflow and optimized with Adagrad optimizer.

4.2 Impacts of Temporal Dynamics and Repeated Patterns (RQ1)

To study the impacts of modeling temporal dynamics and repeated patterns, we compare three variants of TDRB: (1) TDRB_{op} is our proposed TDRB adopting outer product with CNN as the fusion strategy; (2) $\text{TDRB}_{w/o\ rb}$ downgrades TDRB_{op} by merging the repeated behaviors and only keeping the earliest one; (3) $\text{TDRB}_{w/o\ rb\&td}$ is the degraded version of $\text{TDRB}_{w/o\ rb}$, which utilizes a hard rule to order the non-repeated behaviors, that is, *click* → *collect/favorite/bookmark* → *add-to-cart/remove* → *payment/buy/reply*.

The results are demonstrated in Fig. 4, where two major findings can be noted. First of all, the performance of $\text{TDRB}_{w/o\ rb}$ is generally worse than that of TDRB_{op} on the three datasets, implying the benefit of modeling repeated behaviors for a more accurate recommendation. Note that the performance improvements on Taobao2014 and Xing are much larger than those on Taobao2017. This may be attributed to their different distributions w.r.t. the lengths of auxiliary behavior sequence. As shown in Fig. 3, the average length of sequences on Taobao2014 (10) and Xing (3) is longer than that on Taobao2017 (2), Second, by ordering various behaviors via a hard rule, $\text{TDRB}_{w/o\ rb\&td}$ is underperformed by $\text{TDRB}_{w/o\ rb}$ especially on Taobao2017, which validates the assumption that the hard rule cannot truly reflect user behaviors, thus failing to capture the real temporal dynamics. To sum up, the fact that TDRB_{op} achieves the best performance firmly supports the effectiveness of modeling temporal dynamics and repeated behaviors on recommendations.

To further explore the effectiveness of our data segmentation (Sect. 3.1), that is, whether the segmentation will break the original temporal dynamics of the behavior sequences. This is because that a specific target behavior may be not only affected by the auxiliary behaviors in its corresponding segmented sequence $S(u, i)$, and also those behaviors far from it. Hence, we compare TDRB_{op} with TDRB_{pre}, which utilizes all behaviors in previous sequences to help predict the target behavior in the current sequence; As shown in Fig. 5, TDRB_{op} performs

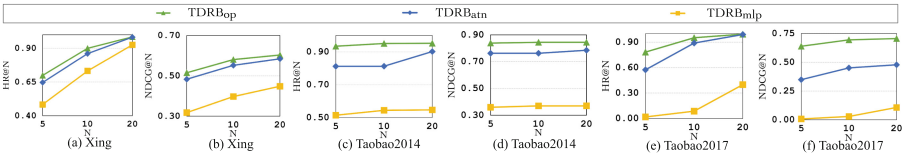


Fig. 5. Impacts of different fusion strategies on TDRB.

comparably to TDRB_{pre} of the three datasets, and even achieves better performance on Taobao2017. This helps verify that: (1) the target behavior is highly affected by the auxiliary behaviors within the same sequence, whilst less influenced by those behaviors far from it; and (2) our data segmentation benefits for accurate recommendation by filtering out potential noise.

4.3 Impacts of Different Fusion Strategies (RQ2)

To further exam the impacts of different fusion strategies on TDRB, three variants are compared: (1) TDRB_{mlp} uses MLP to fuse the influence of target and auxiliary behaviors; (2) TDRB_{atn} employs the attention mechanism to distinguish the importance of target and auxiliary behaviors; (3) TDRB_{op} adopts outer product (with CNN) to integrate target and auxiliary behaviors. Figure 5 depicts the performance on the three datasets. We can observe that TDRB_{mlp} achieves the worst performance in comparison with the other two variants. By automatically distinguishing the saliency of target and auxiliary behaviors, the performance of TDRB_{atn} far exceeds that of TDRB_{mlp}, suggesting the efficiency of attention mechanism. It, however, is outperformed by TDRB_{op}, validating the superiority of both (a) outer product on encoding expressive and high-order user-item correlation; and (b) CNN on capturing the abstract graph features.

4.4 Comparative Results (RQ3)

Table 2 reports the performance of all the comparison methods on the three datasets w.r.t. HR@N and NDCG@N, where $N = \{5, 10, 20\}$. We summarize the major findings as below.

(1) MostPop, as the only non-personalized recommender, performs the worst among all the comparisons across the three datasets, which indicates the essence of personalization in recommendation; (2) The deep learning-based recommenders (e.g., ConvNCF, TDRB) generally outperform the conventional recommenders (e.g., BPRMF, McBPR), demonstrating the superiority of deep learning advances over conventional methods; (3) Regarding to the conventional recommenders, McBPR with the incorporation of both target and auxiliary behaviors performs better than BPRMF on Xing, whilst it underperforms BPRMF on the two Taobao datasets. This might be explained as: the hard rule used to order heterogeneous behaviors in McBPR cannot truly reflect the temporal dynamics of user behaviors, thus introducing noises and hurting the performance. Similar trends can be observed on the performance comparison between the two deep learning-based recommenders: ConvNCF and NMTR. Only target behavior is exploited by ConvNCF; whilst NMTR adopts a hard rule to integrate both target and auxiliary behaviors; (4) Overall, our proposed TDRB achieves the best performance across the three datasets. The significant enhancements, with a lift of 27.91% and 61.32% w.r.t. HR and NDCG on average, demonstrate the effectiveness of modeling temporal dynamics and repeated behaviors.

Table 2. The performance of all comparison methods, where the best performance of the baselines is marked with '*'; the performance of TDRB (i.e., TDRB_{op} that performs the best among all variants of TDRB) is highlighted in bold; 'Improve' indicates the improvements that TDRB achieves relative to the '*' results.

Datasets	Methods	HR@5	NDCG@5	HR@10	NDCG@10	HR@20	NDCG@20
Xing	MostPop	0.0021	0.0021	0.0065	0.0035	0.0065	0.0035
	BPRMF	0.4203	0.2603	0.6928	0.3483	0.8213	0.3824
	McBPR	0.4204*	0.2607*	0.7037*	0.3518*	0.8714*	0.3953*
	ConvNCF	0.4117	0.2545	0.6492	0.3309	0.8257	0.3766
	NMTR	0.4031	0.2546	0.4749	0.2777	0.5882	0.3063
	TDRB	**0.7058**	**0.5467**	**0.8845**	**0.6043**	**0.9825**	**0.6304**
	Improve	70.92%	109.70%	25.69%	71.77%	12.86%	59.47%
Taobao 2014	MostPop	0.0057	0.0031	0.0229	0.0081	0.0229	0.0081
	BPRMF	0.6533	0.4595	0.6832	0.4699	0.6851	0.4704
	McBPR	0.6564	0.4552	0.6908*	0.4670	0.6953*	0.4681
	ConvNCF	0.6596*	0.4782*	0.6838	0.4865*	0.6870	0.4873*
	NMTR	0.5876	0.4135	0.6195	0.4244	0.6214	0.4249
	TDRB	**0.9343**	**0.8385**	**0.9509**	**0.8441**	**0.9521**	**0.8444**
	Improve	41.64%	75.34%	27.35%	73.50%	36.93%	35.71%
Taobao 2017	MostPop	0.0233	0.0131	0.0425	0.0186	0.0425	0.0186
	BPRMF	0.5879	0.3951	0.8852	0.4914	0.9949*	0.5206
	McBPR	0.5772	0.3818	0.8859	0.4815	0.9947	0.5104
	ConvNCF	0.6066*	0.4127*	0.8863*	0.5033*	0.9944	0.5321*
	NMTR	0.4338	0.2660	0.6930	0.3499	0.8533	0.3910
	TDRB	**0.7826**	**0.6391**	**0.9520**	**0.6945**	**0.9944**	**0.7058**
	Improve	29.01%	55.85%	6.90%	37.98%	−0.05%	32.64%

5 Conclusions and Future Work

This paper proposes a novel neural framework TDRB, which exploits the temporal dynamics and repeated behaviors to further enhance the accuracy of heterogeneous implicit feedback based recommendation. Being equipped with three core modules (i.e., `Target`, `Auxilary` and `Fusion`), TDRB is capable of better modeling user preference, thus achieving superior recommendation performance. Extensive empirical studies on three real-world datasets validate the effectiveness of our proposed TDRB. For future work, we intend to explore more values of the repeated behaviors for the next item recommendation.

Acknowledgments. This work is partially supported by the National Natural Science Foundation of China under Grant No. 61972078 and 61702084, and the Fundamental Research Funds for the Central Universities under Grant No. N181705007.

References

1. Ding, J., et al.: Improving implicit recommender systems with view data (2018)
2. Gao, C., et al.: Neural multi-task recommendation from multi-behavior data (2019)
3. Guo, G., et al.: Resolving data sparsity by multi-type auxiliary implicit feedback for recommender systems. Knowl.-Based Syst. **138**, 202–207 (2017)
4. He, K., et al.: Deep residual learning for image recognition. In: CVPR (2016)
5. He, X., et al.: Fast matrix factorization for online recommendation with implicit feedback. In: SIGIR (2016)
6. He, X., et al.: Neural collaborative filtering (2017)
7. He, X., et al.: Outer product-based neural collaborative filtering (2018)
8. Hidasi, B., et al.: Session-based recommendations with recurrent neural networks. arXiv: Learning (2015)
9. Li, J., et al.: Neural attentive session-based recommendation. arXiv: Information Retrieval (2017)
10. Loni, B., et al.: Bayesian personalized ranking with multi-channel user feedback. In: RecSys (2016)
11. Pan, W., et al.: GBPR: group preference based Bayesian personalized ranking for one-class collaborative filtering. In: IJCAI (2013)
12. Pan, W., et al.: Adaptive Bayesian personalized ranking for heterogeneous implicit feedbacks. Knowl.-Based Syst. **73**, 173–180 (2015)
13. Qiu, H., et al.: BPRH: Bayesian personalized ranking for heterogeneous implicit feedback. Inf. Sci. **453**, 80–98 (2018)
14. Qiu, H., et al.: TBPR: trinity preference based Bayesian personalized ranking for multivariate implicit feedback. In: UMAP (2016)
15. Rendle, S., et al.: BPR: Bayesian personalized ranking from implicit feedback. In: UAI (2009)
16. Wang, C., et al.: Modeling item-specific temporal dynamics of repeat consumption for recommender systems. In: WWW (2019)
17. Wen, H., et al.: Leveraging multiple implicit feedback for personalized recommendation with neural network. In: AIAM (2019)
18. Yin, H., et al.: SPTF: a scalable probabilistic tensor factorization model for semantic-aware behavior prediction. In: ICDM (2017)
19. Young, T., et al.: Recent trends in deep learning based natural language processing (2018)