2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Behavior-Type Aware Representation Learning for Multiplex Behavior Recommendation

Xin-Wei Yao[†*1], ShiXun Sun[†1], Chuan He[1], Xin-Li Xu[1], Wei Huang[1], Qiang Li[2], Xinggang Fan[3]

[1] College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China
[2] Institute for Frontier and Interdisciplinary Sciences, Zhejiang University of Technology, Hangzhou, China
[3] College of Zhijiang, Zhejiang University of Technology, Shaoxing, China
{xwyao, sunshixun, hechuan, xxl, huangwei, qiangli, xgfan}@zjut.edu.cn

Abstract. Efficient recommender systems are essential for modeling user-item interactions, such as views, favorites, and purchases. However, two challenges remain: 1) Effectively modeling complex multiplex behavior patterns derived from user-item interactions necessitates more sophisticated representation learning techniques. 2) Existing methods often neglect the differential impact of various auxiliary interactions on the primary target interaction (e.g., purchase). In this study, we propose a more informative framework, Behavior-Type Aware Representation Learning for Multiplex Behavior Recommendation (BA-MBRec), to learn representations of users and items by mining behavior-aware patterns in feature encoding. Specifically, BA-MBRec is a powerful approach tailored to effectively encode nodes across various multiplex structures. It not only adaptively captures individual behavior-aware patterns but also discovers the interdependencies across these various patterns within multiplex heterogeneous networks by hierarchical modeling and cross-behavioral aggregators. Experiments on three real-world datasets demonstrate its superior performance, with improvements of 5.2% in HR@10 and 10.16% in NDCG@10 over state-of-the-art methods. Our empirical studies further demonstrate the great potential of this framework for capturing the multiplexity of users' preferences in recommendation scenarios. Our implementation code is available in https://github.com/sunshixx/BA-MBRec/tree/master.

Keywords: Recommender systems, Multiplex Heterogenous Graph, Learning latent representations, Contrastive Learning.

1 Introduction

Recommender systems are crucial for mitigating information overload by providing personalized suggestions in domains like retail, advertising, and social media [1]. Graph Neural Networks (GNNs) have emerged as a powerful tool for this task, adept at processing graph-structured data and learning high-quality user/item representations through aggregating higher-order information from neighboring nodes. Based on this,

† These authors contributed equally to this work and should be considered co-first authors.

* Corresponding authors.

many methods make many efforts on representation learning for homogeneous networks with a single type of node [2][3].

However, traditional GNNs, designed for homogeneous graphs, struggle with the diverse user behaviors common in E-commerce, such as viewing, favoriting, adding to cart, and purchasing. Consequently, subsequent studies have proposed multi-behavior recommender systems, which attempt to meticulously consider multiplex relations between users and items by introducing meta-paths [5], contrastive learning [4], and other means [6][7]. Nevertheless, existing approaches often struggle to fully capture the intricate impact that user-item interactions, involving multiplex relations, have on the learned user and item representations. In response, some works [8][9] explore behavior patterns, precisely modeling the dynamic and personalized influence of different interactions within complex behavior patterns on the target relation, and effectively distinguishing their relative importance, remains an area requiring further investigation.

1) Comprehensively exploiting multiplex behavior is challenging. Comprehensively exploiting multiplex behavior is challenging. Current strategies broadly fall into two categories: learning from individual behavior subgraphs before integration, or applying GNNs directly to the overall heterogeneous graph. However, both approaches often struggle to adequately model the intricate patterns and dependencies within multiplex interactions. Recognizing this, methods [8][9] have explored explicit behavior pattern or relation chain modeling. While effective in capturing predefined structures, defining and managing these explicit patterns can be complex, and they may not flexibly encompass all relevant high-order interaction sequences present in the data. Furthermore, alternative strategies relying on explicitly defined interaction paths (e.g., meta-paths) face significant scalability hurdles due to the combinatorial explosion of possible paths. These combined limitations highlight the urgent need for more sophisticated and scalable approaches to multi-behavior representation learning.

2) Feature aggregation must account for multifaceted interactions. Each behavior type contributes uniquely to predicting the target behavior in recommendation systems. Different behaviors not only provide complementary knowledge for understanding user interests but also interact in intricate ways. However, existing methods often struggle to fully capture these varying impacts. The difficulties include (a) dynamically discerning the personalized importance of different concurrent behaviors during aggregation at various interaction depths, and (b) explicitly modeling the personalized influence exerted by auxiliary behaviors specifically on the target behavior for individual users. While inspired by Transformers, various ways are developed to effectively model behavior heterogeneity and dependencies [10]. However, methods that can adaptively capture finer-grained relationships and personalized modeling of target-auxiliary relationships are still needed.

To address these challenges, we propose a novel framework, Behavior-Type Aware Representation Learning for Multiplex Behavior Recommendation (BA-MBRec). We design a behavior-aware pattern encoder that implicitly models intricate, multi-hop behavior patterns via layer-wise message passing, thereby capturing rich structural and semantic information while avoiding the scalability issues of explicit path enumeration. This encoder is enhanced by a hierarchical behavioral dependency modeling component, designed to capture nuanced semantic similarities and dependencies among different behavior types at the same interaction depth. To ensure feature aggregation accounts for multifaceted interactions, we design a cross-behavioral aggregation that adaptively weighs and fuses representations derived from diverse behavior types and patterns, providing a more fine-grained and personalized aggregation than static or uniform approaches. Additionally, BA-MBRec leverages contrastive learning to adaptively model the relationship between the target behavior and auxiliary behaviors. By utilizing information from auxiliary behaviors, the model enriches the representation of the target behavior, improving its ability to discern the varying influence of different behaviors on the target behavior and account for the heterogeneity of user preferences. BA-MBRec provides a more robust and nuanced approach to multi-behavior recommendation. The subsequent sections detail the architecture and empirical validation of our framework.

Briefly, the contributions of this work could be concluded as follows:

- We propose a novel multi-behavior recommendation model for multiplex heterogeneous scenarios, which tackles the challenges of behavior complexity and explores the node embedding based on users' preferences.
- We develop an implicit behavior pattern encoder via multiplex convolution, enhanced by hierarchical dependency modeling and cross-behavior aggregation to adaptively extract high-order semantic patterns and dependencies, further complemented by meta-weighted contrastive learning to personalize behavior influence modeling.
- We conduct experiments on three real-world recommender datasets to demonstrate the effectiveness of our proposed model. By comparing with 10 STOA baselines, our model could be improved up to 5.2% and 10.16% in **HR@10** and **N@10**.

2 Related Work

2.1 Graph-based Recommender Models

The ability of Graph Neural Networks (GNNs) to model complex relationships within graph structures has led to promising results in recent recommender models. These models effectively leverage various propagation functions to aggregate embeddings from neighboring nodes. For example, NGCF [3] propagates embeddings through user-item graphs to explicitly model high-order collaborative signals. LightGCN [2] improves recommendation by simplifying NGCF's message passing via sum-based aggregation without weight matrices. GTNs [11] learn new graph structures and node representations by identifying useful connections, guided by task objectives. Inspired by these works, our model employs a GNN architecture to capture high-order information within heterogeneous paths, leveraging the user-item graph structure for improved recommendation accuracy.

2.2 Multi-behavior Recommendation

Considering the various types of user-item interactions, recent studies have explored effective methods for addressing behavior multiplicity. MATN [12] incorporates attentive weights for pattern aggregation, and MB-GMN [13] combines meta-learning with multi-behavior messages to explore low-rank behavioral embeddings. MBGCN [14] employs graph convolutional networks for discriminative behavior representation. Approaches such as [15], and MHGCN [7] leverage stacked convolutions for heterogeneous network learning. However, these methods often oversimplify multiplex structures, neglecting high-level semantics. To address this, our work proposes a sophisticated architecture that adaptively models interactions and dependencies among diverse user behaviors, enhancing representation learning in multiplex networks.

3 Preliminary

A multiplex heterogenous graph is defined $\mathcal{G} = \{\mathcal{U}, \mathcal{I}, \mathcal{E}\}$. Here, $\mathcal{U} = \{u_1, \cdots, u_N\}$ and $\mathcal{I} = \{v_1, \cdots, v_J\}$ represent the set of users and items, respectively. $\mathcal{E} = \bigcup_{r \in \mathcal{R}} \mathcal{E}_r$ represents the collection of various interactive edges between users and items, each edge belonging to a particular interaction type. $\mathcal{R}$ denotes the set of all interaction types. In our multi-behavior scenario, we let A_r denote the user-item interaction matrix under the r-th behavior type (e.g., view, cart, favorite, and buy in the commercial recommendation scenario). Then the observed user-item interaction with various types of behaviors could be present as follows: $\{A_1, \cdots, A_{rt}, \cdots, A_{|\mathcal{R}|}\}$, where rt represents the target type of behavior. From the Gross Merchandise Value (GMV) perspective, purchase is always considered the target behavior in E-commercial services.

3.1 User-Item interaction matrix

Since there are various types of behavior in user-item interactions, a certain type of interaction matrix is defined as A_r. Specifically, the individual element $x_{i,j}^r \in A_r$ is set as 1 if the user u_i interacts with the item v_j under the r-th behavior type, $x_{i,j}^r = 0$ otherwise.

3.2 Multiplex interaction matrix

Inspired by the representation paradigm of Heterogenous Graph Convolutional Network [7], we explore a matrix stacking method to compose all user-item interactions $\{A_1, \cdots, A_{rt}, \cdots, A_{|\mathcal{R}|}\}$ into one multiplex interaction matrix $\mathbb{A}$. This matrix provides a structured way to represent the multiple layers or types of interactions between users and items. Specifically, $x_{i,j} \in \mathbb{A}$ is set as 1 if there exists any interaction between the user u_i and the item v_j, $x_{i,j} = 0$ otherwise.

3.3 Behavior Aware Pattern, or BAP

A two hop behavior pattern such as $User \xrightarrow{click} Item \xrightarrow{buy} User$ and $User \xrightarrow{click} Item \xrightarrow{favorite} User$ both represent a broader behavioral dependency between users and items. This User-Item-User relationship, starting with *click* behavior, can be seen as a behavior-aware pattern [click] $\circ$ [·], where $\circ$ denotes the composition operator on relations, and [·] denotes optional.

3.4 Problem Statement

Given these definitions, we can now define the task as:

- **Input:** The observed multiplex heterogenous graph $\mathcal{G} = \{\mathcal{U}, \mathcal{I}, \mathcal{E}\}$ is given.
- **Output:** A predictive function that can capture rich structural and semantic information involved in the heterogenous graph for estimating the likelihood that the user $u \in \mathcal{U}$ will interact with the item $v \in \mathcal{I}$ under the target type rt of behaviors.

4 Methodology

In this section, we will elaborate on the workflow of Behavior-Type Aware Representation Learning for Multiplex-Behavior Recommendation **BA-MBRec**, as illustrated in **Fig. 1**.

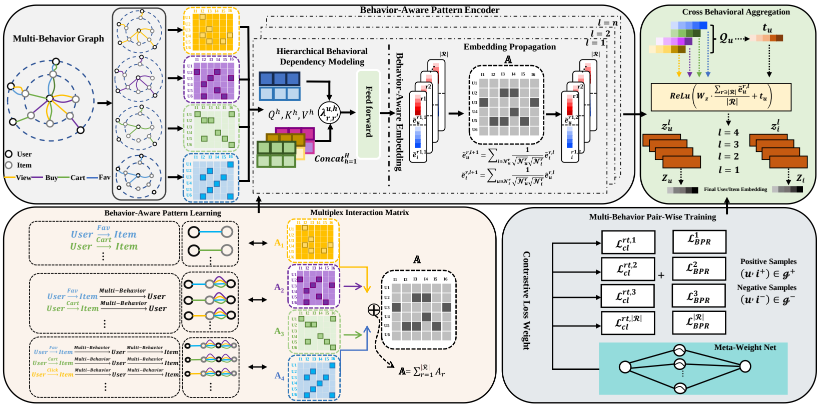

Fig. 1. The framework of BA-MBRec, best viewed in color.

4.1 Overall of Behavior-Aware Encoder

This section details the BA-MBRec framework, designed to effectively model multi-behavior interactions for recommendation. As highlighted in the Introduction, effectively capturing the rich information embedded in multiplex behaviors is non-trivial. Traditional strategies relying on explicit path definition (e.g., meta-paths) suffer from exponential complexity. To overcome this, our Behavior-Aware Encoder adopts an implicit modeling strategy based on layer-wise graph message passing. This allows the model to capture multi-hop dependencies and synergistic interactions between diverse behavior types without explicit path enumeration. The encoder operates through three core stages detailed below: 1) Automatic construction of representations for behavior-aware patterns of varying lengths. 2) Hierarchical Behavioral Dependency Modeling at each hop to capture semantic relationships among patterns of the same length. 3) Information propagation and aggregation to learn final node embeddings.

Behavior Pattern Construction. As mentioned in the preliminary, the corresponding user-item interaction matrix $\{A_1, \cdots, A_{rt}, \cdots, A_{|\mathcal{R}|}\}$ could be used to describe subgraphs that correspond to individual types of basic behaviors, where $A_r \in \mathbb{R}^{(|\mathcal{V}| \times |\mathcal{I}|)}$. Then, each user-item interaction matrix could be operated to generate the multiplex interaction matrix $\mathbb{A}$ as follows:

$$\mathbb{A} = \sum_{r=1}^{|\mathcal{R}|} A_r \tag{1}$$

Here, the variable in the matrix $\mathbb{A}$ represents the aggregated interaction matrix summing interactions across all behaviors. The behavior-aware pattern $\mathcal{M}_r^{(l)}$ reflecting interactions starting with behavior r and extending for l hops, can be obtained via matrix multiplication:

$$\begin{aligned} \mathcal{M}_r^{(l)} &= A_r^{(1)} \mathbb{A}^{(2)} \cdots \mathbb{A}^{(l)} \\ &= A_r^{(1)} \left(\textstyle\sum_{r=1}^{|\mathcal{R}|} A_r\right)^{(2)} \cdots \left(\textstyle\sum_{r=1}^{|\mathcal{R}|} A_r\right)^{(l)} \end{aligned} \tag{2}$$

This formulation allows the encoder to capture the influence of multi-hop paths initiated by a specific behavior type r. By varying the initial matrix $A_r^{(1)}$ and the number of subsequent multiplications with $\mathbb{A}$ (representing one hop considering all behaviors), we can generate representations for patterns of arbitrary length l starting from any specific behavior r. These matrices $\mathcal{M}_r^{(l)}$ effectively encode the reachability and structure of specific behavior-aware patterns.

Hierarchical Behavioral Dependency Modeling. In recommendation scenarios, different types of user behaviors (e.g., view, cart, buy), even when occurring within patterns of the same length (i.e., represented at the same hop distance l in the GNN), may exhibit fine-grained interdependencies and semantic similarities. For instance, for a level-headed user, 'cart' behavior might be semantically closer to 'view' than 'buy', when considering patterns of length l. Capturing these relationships within each layer is crucial for a nuanced understanding of user preferences.

To model these intra-layer dependencies, we propose the hierarchical behavioral dependency modeling. Crucially, it operates at each graph propagation layer l. It takes the set of behavior-specific user embeddings learned at that layer, $\{e_u^{r,l} | r \in \mathcal{R}\}$, as input.

Specifically, during every propagation step, it evaluates the correlations $\lambda_{r,r'}^{u,h}$ between different pattern types at the same pattern length l:

$$\tilde{e}_u^{r,l} = \text{MH-Att}(\mathbf{e}_u^{r,l}) = \underset{h=1}{\overset{H}{\text{Concat}}} \left(\sum_{r'=1}^{|R|} \lambda_{r,r'}^{u,h} \cdot \mathbf{V}^h \cdot \mathbf{e}_u^{r,l}\right)$$
$$\lambda_{r,r'}^{u,h} = \frac{\exp \bar{\lambda}_{r,r'}^{u,h}}{\sum_{r'=1}^{|R|} \exp \bar{\lambda}_{r,r'}^{u,h}}; \bar{\lambda}_{r,r'}^{u,h} = \frac{(\mathbf{Q}^h \cdot \mathbf{e}_u^{r,l})^\top (\mathbf{K}^h \cdot \mathbf{e}_u^{r',l})}{\sqrt{d/H}} \tag{3}$$

Here $\lambda_{r,r'}^{u,h}$ represents the learned attention weight signifying the relevance of behavior r' to behavior r for user u within the h-th attention head, specifically at layer/hop l. Where $\mathbf{Q}^h$, $\mathbf{K}^h$ and $\mathbf{V}^h$ are learnable projection matrices of h-th head learning sub-space. $\tilde{e}_u^{r,l}$ is a refined embedding for behavior type r at length l, enriched with context from other concurrent behaviors at that same length. This process allows the model to learn nuanced similarities and dependencies (e.g., [click] ∘ [·] vs. [view] ∘ [·] at hop l) without conflating information across different pattern lengths. Similar operations are applied for item embeddings.

Information Propagation. The core propagation mechanism utilizes a chain of LightGCNs [2]. After refining the behavior-specific embeddings at layer l using HBDM to obtain $\{\tilde{e}_u^{r,l}, \tilde{e}_i^{r,l}\}$, these enhanced representations contribute to the integration process, which could be demonstrated as follows:

$$\begin{aligned} \tilde{e}_u^{r,l+1} &= \sum_{i \ni \mathcal{N}_u^r} \frac{1}{\sqrt{|\mathcal{N}_u^r|}\sqrt{|\mathcal{N}_i^r|}} \tilde{e}_i^{r,l} \\ \tilde{e}_i^{r,l+1} &= \sum_{u \ni \mathcal{N}_i^r} \frac{1}{\sqrt{|\mathcal{N}_u^r|}\sqrt{|\mathcal{N}_i^r|}} \tilde{e}_u^{r,l} \end{aligned} \tag{4}$$

where $\mathcal{N}_u^r$ and $\mathcal{N}_i^r$ are neighbors defined by the behavior pattern matrix $\mathcal{M}_r^{(l)}$.

4.2 Cross behavioral Aggregation

To address the challenge of effectively aggregating features while accounting for multifaceted interactions, we propose the cross-behavioral aggregation. Firstly, we concatenate all the embeddings for node u as $\mathcal{Q}_u \in \mathbb{R}^{d \times |\mathcal{R}|}$, where d is the dimension of refined behavior-aware embeddings at layer l (similar aggregation is applied for item nodes) :

$$\mathcal{Q}_u^{(l)} = (\tilde{e}_u^{1,l}, \tilde{e}_u^{2,l}, \cdots, \tilde{e}_u^{|R|,l}) \tag{5}$$

Then we use the self-attention mechanism to compute personalized, dynamic weights $\beta_{u,r} \in \mathbb{R}^{|\mathcal{R}|}$ of embeddings in $\mathcal{Q}_u^{(l)}$ on behavior-aware pattern r as:

$$\beta_{u,r}^{(l)} = softmax(w_r^T \tanh(W_r Q_u^{(l)}))^T = (\frac{\exp(w_r^T \tanh(W_r \tilde{e}_u^{r,l}))}{\sum_{r\in|\mathcal{R}|} \exp(w_r^T \tanh(W_r \tilde{e}_u^{r,l}))})^T \tag{6}$$

where w_r and W_r are trainable parameters for behavior type r with size $d_a \times d$ respectively. The superscript T denotes the transposition of the vector or the matrix. These weights are used to perform a weighted aggregation, yielding a fused embedding $t_u^{(l)}$ under the multiplex behavior view:

$$t_u^{(l)} = Q_u^{(l)} \beta_{u,r}^{(l)} \tag{7}$$

After performing weighted aggregation under different behavior patterns, we suggest supplying base information for feature learning as follows:

$$z_u^l = ReLu\left(W_z \cdot \frac{\sum_{r\in|\mathcal{R}|} \tilde{e}_u^{r,l}}{|\mathcal{R}|} + t_u^{(l)}\right) \tag{8}$$

However, in some applications, both long and short behavior patterns are essential\cite{yun2019graph}. To learn both short and long behavior-aware patterns, including original behavior information, we structure the embedding Z_u in different lengths as:

$$Z_u = f_\theta(z_u^1, z_u^2, \dots, z_u^l) \tag{9}$$

where f_θ involves concatenation followed by a linear transformation. W_z is a learnable matrix parameter.

4.3 Joint Optimization

Behavioral Contrastive Loss. Based on the target behavior (typically "buy") and other auxiliary behaviors we mentioned above, BA-MBRec adopts contrastive learning on different behavior representations, which treats the different behaviors for the same user as positive sample pairs and different users as negatives. Following the works [16][17], we utilize the InfoNCE loss in our cross-behavior contrastive view to measure the distance between embeddings by maximizing the contrastive loss, which enforces the agreement between two behaviors of the same user and divergence among different users:

$$\mathcal{L}_{cl}^{rt,r} = \sum_{u\in\mathcal{U}} -\log \frac{\exp(\varphi(e_u^{rt}, e_u^r)/\tau)}{\sum_{u'\in\mathcal{U}} \exp(\varphi(e_u^{rt}, e_{u'}^r)/\tau)} \tag{10}$$

Where $\varphi(\cdot)$ denotes the similarity function (e.g., inner-product or cosine similarity) between two embeddings. τ represents the temperature hyperparameter for the softmax function. After that, we can obtain the contrastive $\mathcal{L}_{cl}^{rt,r}$ for each pair of target behavior rt and auxiliary behavior r.

BPR Loss. BA-MBRec employs the Bayesian Personalized Ranking (BPR) recommender loss as one of the task losses for the next parameter optimization:

$$\mathcal{L}_{BPR}^{r} = \sum_{(u,i^+,i^-)\in\mathcal{O}_r} -\ln\left(\sigma\left(\hat{x}_{u,i^+}^{r} - \hat{x}_{u,i^-}^{r}\right)\right) + \lambda\|\Theta\|^2 \tag{11}$$

Where $\hat{x}_{u,i^+}^{r} = e_u^{r\mathrm{T}} \cdot e_{i^+}^{r}$, O_r denotes training pairwise samples of r-th behavior-aware pattern (i.e., $\mathcal{O}_r = \{(u,i^+,i^-)|(u,i^+) \in g^+, (u,i^-) \in g^-\}$), g^+ and g^- represents the observed and unobserved interaction of specific user under the corresponding behavior-aware pattern. Θ denotes the learnable parameters and L_2 regularization is used to alleviate the overfitting issue. λ a coefficient to control the L_2 regularization.

Meta-Weight Encoder. Following the meta-knowledge integration in CML [4] and DCMGNN [9], we adopted a meta-knowledge encoder based on learned user representation Z_u and e_u^r:

$$\begin{aligned} M_{u,1}^{rt,r} &= \left(dup\left(\mathcal{L}_{cl}^{rt,r}\right)\cdot\gamma\right) \parallel e_u^r \parallel \mathcal{Z}_u \\ M_{u,2}^{rt,r} &= \mathcal{L}_{cl}^{rt,r} \cdot (e_u^r \parallel \mathcal{Z}_u) \end{aligned} \tag{12}$$

Where rt denotes the target behavior and $M_{u,1}^{rt,r}$, $M_{u,2}^{rt,r}$ encodes meta-knowledge between behavior patterns and the target behavior. $dup(\cdot)$ duplicates to match the embedding dimension. $\parallel$ denotes the concatenation operation. γ is a scale factor. This encoded meta-knowledge then informs weights for different contrastive losses, enabling customized capture of personalized user preferences based on multi-behavior interactions. Finally, the personalized contrastive loss weight could be represented as follows:

$$\omega_{cl}^{rt,r} = \omega_{u,1}^{rt,r} + \omega_{u,2}^{rt,r} = \xi\left(M_{u,1}^{rt,r}\right) + \xi(M_{u,2}^{rt,r}) \tag{13}$$

where $\xi(\cdot)$ represents network with LeakyReLU. Subsequently, we can apply our weighting scheme to various task losses, including the behavioral contrastive loss and the BPR-based recommendation objective loss:

$$\begin{aligned} \mathcal{L}_{cl} &= \sum_{r\in\mathcal{R}} \omega_{cl}^{rt,r}\mathcal{L}_{cl}^{rt,r} \\ \mathcal{L} &= \mu\mathcal{L}_{cl} + \sum_{r\in\mathcal{R}} \omega_{bpr}^{rt,r}\mathcal{L}_{BPR}^{r} \end{aligned} \tag{14}$$

4.4 Model Complexity Analysis.

We estimate the computational complexity of our model by considering the following parts: (i) The computational cost of the LightGCN architecture adopted by our model is $O(L \times K \times d \times |R_k|)$ with L layers of network and K types of behaviors. $|R_k|$ represents the number of edges in the behavior-aware pattern matrix under K behavior. For hierarchical behavioral dependency modeling, the most prominent computation comes from the $O(K \times d^2 \times (N + M))$. The operation of linear transformation and cross-typed behavioral aggregation asymptotically takes $O(L \times (N + M) \times d \times (K + d))$. (ii) InfoNCE loss computation costs $O(B \times d)$ and $O(B \times S \times d)$ for the numerator and denominator as shown in Eq.10. Here S indicates the sampled set for considering time complexity and robustness at the same time [16]. Therefore, the contrastive paradigm takes $O(K \times |R_k| \times S \times d)$ per epoch over training. BA-MBRec maintains comparable time complexity while improving performance [4][18].

5 EVALUATION

To evaluate BA-MBRec's performance, we conduct experiments on three different real-world datasets and compare them with state-of-the-art multi-behavior recommendation techniques. Particularly, we propose the following research questions:

- **RQ1:** How effectively does BA-MBRec architecture work in tackling multi-behavior recommendations?
- **RQ2:** How do the different components impact the effectiveness of the BA-MBRec framework for multi-behavior recommendations?
- **RQ3:** How does BA-MBRec make recommendations based on different user behaviors?
- **RQ4:** How does BA-MBRec perform with varying interaction sparsity degrees?
- **RQ5:** What is the impact of varying key hyperparameters on the performance of BA-MBRec?
- **RQ6:** How is the explainability of BA-MBRec? How effectively is the BA-MBRec capturing behavior-aware patterns for the final recommendation tasks?

5.1 Experimental Setting

Dataset. We perform our model BA-MBRec on three real-world public datasets demonstrated in **Table 1**. Taobao-Data: This dataset from Taobao, one of the largest e-commerce platforms in China, includes four types of user-item interactions: page view, add-to-cart, tag-as-favorite, and purchase. IJCAI-Contest: This dataset originates from the IJCAI-15 Challenge and is derived from a business-to-customer retail system. It includes the same types of user behaviors as the Taobao dataset, representing different user intentions towards items. Retailrocket: This dataset is collected from the Retailrocket recommendation system. It includes user interactions such as Page View, Add-to-Cart, and Transaction. We consider purchasing a target behavior, with other types serving as auxiliary behaviors following the existing work.

Table 1. Statistical information of evaluation datasets.

Dataset	Users	Items	Interactions	Behavior Types
Taobao	147,894	99,037	7,658,926	{PageView, Fav, Cart, Buy}
IJCAI	17,435	35,920	799,368	{Click, Fav, Cart, Buy}
Retail	2,174	30,113	97,381	{PageView, Cart, Transaction}

Baseline. We divided state-of-the-art techniques into three groups:

Conventional Matrix Factorization Approach:

- **BPR [19]:** A popular matrix factorization model that uses Bayesian personalized ranking as its optimization criterion.

Graph Neural Networks for Recommendation:

- **NGCF [3]:** Convolutional message-passing on user-item graph to capture high-order collaborative effects in user embeddings.
- **LightGCN [2]:** A simplified and efficient graph convolutional network designed for collaborative filtering.

Multi-behavior Models for Recommendation:

- **MBGCN [14]:** The GCN-based model uses a consolidated graph of user interactions to capture multi-behavioral patterns via behavior-aware embedding propagation.
- **MATN [12]:** MATN uses memory-enhanced self-attention for multi-behavior recommendation, measuring inter-behavior influence.
- **KHGT [10]:** Transformer-based temporal multi-behavior modeling with graph attention differentiation.
- **MHGCN [7]:** MHGCN proposes a Multiplex Heterogeneous Graph Convolutional Network that captures heterogeneous metapath interactions and integrates structural signals with attribute semantics.
- **BPHGNN [8]:** BPHGNN proposes a Behavior Pattern-based Heterogeneous Graph Neural Network that captures multiplex structural signals and global information through depth and breadth behavior pattern aggregation.
- **CML [4]:** Multi-behavior contrastive learning framework distilling knowledge across behaviors via contrastive loss.
- **KMCLR [18]:** Multi-behavior learning for user embeddings, enhanced by knowledge graphs for robust item representations.

Evaluation Protocols. Hit Ratio **(HR@N)** and Normalized Discounted Cumulative Gain **(NDCG@N)** are applied to evaluate the performance of our model in the recommendation task. We set all baseline models for fair comparison according to CML's experimental setup. Specifically, in the test data, we used the last interaction item of the predicted behavior as a positive example and 99 randomly selected items that the user did not interact with as negative examples.

Parameter Setting. We implement our model BA-MBRec in Pytorch. All embeddings were initialized with Xavier [22], and the model was optimized using the AdamW optimizer [23] and the Cyclical Learning Rate (CyclicLR) strategy [24], which systematically vary the learning rate within predefined bounds during training, helping the model escape local minima and improve generalization. Following the existing work [4], we set the base and max learning rate in $\{0.6e^{-4}, 1e^{-4}, 1e^{-3}\}$ and $\{0.6e^{-3}, 1e^{-3}, 2e^{-3}, 5e^{-3}\}$, respectively. We also use dropout to alleviate the problem of overfitting in Meta-Weight Net.

Table 2. Performance comparison on different datasets in terms of HR@10 and NDCG@10.

Dataset	Metric	BPR	LightGCN	NGCF	MBGCN	MATN	KHGT	MHGCN	BPHGNN	CML	KMCLR	Ours	Im-prov.%
Taobao	**HR**	0.2436	0.3419	0.3354	0.4060	0.4276	0.4319	0.4530	0.4901	0.5782	<u>0.5906</u>	**0.6153**	**4.18%**
	NDCG	0.1465	0.2049	0.1902	0.2195	0.2543	0.2677	0.2891	0.3199	0.3594	<u>0.3678</u>	**0.3945**	**7.26%**
IJCAI	**HR**	0.1765	0.2695	0.2713	0.3760	0.3710	0.3964	0.4230	0.4785	0.4905	<u>0.5234</u>	**0.5506**	**5.20%**
	NDCG	0.0891	0.1261	0.1281	0.1910	0.2432	0.2399	0.2577	0.2820	0.2871	<u>0.3169</u>	**0.3491**	**10.16%**
Retail	**HR**	0.2479	0.2828	0.2814	0.3108	0.3010	0.3719	0.3482	0.4003	0.3908	<u>0.4557</u>	**0.4723**	**3.65%**
	NDCG	0.1557	0.1683	0.1622	0.1857	0.1997	0.2242	0.2379	0.2402	0.2361	<u>0.2735</u>	**0.2922**	**6.84%**

5.2 Performance Comparison (RQ1)

Table 2 displays the Top-N item recommendation results for the target behavior type, obtained from comparable experiments on three real-world datasets. The *"improv."* denotes the relative improvement between the baselines and our model BA-MBRec. BA-MBRec outperforms other techniques on each dataset, demonstrating our model's effectiveness in modeling personalized multi-behavior patterns. We reckon the following factors are the primary drivers behind the significant improvement: (i) Benefiting from the techniques of the behavior-aware pattern encoder, the model could fully dig into high-order semantic information from varying behavior patterns. (ii) Introducing the proposed attention mechanism allows the model to weightedly aggregate embeddings of multiplex behaviors. (iii) Contrastive learning provides informative gradients to graph-based collaborative filtering, improving preference exploration and leading to more accurate recommendations.

The multi-behavior recommendation always outperforms the single-behavior recommendation, which indicates the effectiveness of introducing auxiliary information about other behavior types. These graph-based methods effectively model complex user-item relationships and capture higher-order interactions, leading to impressive results.

5.3 Ablation Study (RQ2)

We further conduct the ablation study for our BA-MBRec to evaluate each components' effectiveness in **Table 3**. The details of our experiments are as follows:

Table 3. Results of ablation experiments.

Dataset	**Taobao**		**IJCAI-Contest**		**Retailrocket**	
Metrics	**HR**	**NDCG**	**HR**	**NDCG**	**HR**	**NDCG**
BA-MBRec(Decoupled)	0.5945	0.3817	0.5307	0.3280	0.4608	0.2701
BA-MBRec(GlobalMeanPooling)	0.5989	0.3826	0.5158	0.3138	0.4610	0.2778
BA-MBRec(LinearProj)	0.5911	0.3791	0.5217	0.3255	0.450	0.2635
BA-MBRec(UniformCL)	0.5862	0.3805	0.5092	0.3086	0.4324	0.2535
BA-MBRec(NoCL)	0.5959	0.3824	0.5240	0.3231	0.4591	0.2574
BA-MBRec	0.6126	0.3945	0.5506	0.3491	0.4723	0.2922

- **Effect of Behavior-aware Pattern Construction:** We create a model variant known as BA-MBRec(Decoupled) by turning off the construction of topological behavioral patterns. Alternatively, we utilize graph neural networks to capture basic behavior information and train the model on a decoupled view of behaviors. Notably, our model obtains better performance than BA-MBRec(Decoupled), which demonstrates a heightened capacity to characterize behavior embeddings accurately and effectively, as it autonomously captures high-order behavioral pattern information.
- **Effect of Cross-Behavioral Aggregation:** To isolate the contribution of the attention-based fusion mechanism within Cross-Behavioral Aggregation (CBA), we designed the BA-MBRec(GlobalMeanPooling) variant. In this variant, the attention mechanism used for dynamically weighting behavior embeddings is removed. As an alternative, we adopt a fine-grained non-adaptive aggregation strategy, which stacks user and item embeddings generated under different behavioral patterns into a multi-channel representation and aggregates the multi-channel information through global mean pooling. As a result, BA-MBRec exhibits a superior ability to capture the dependencies among diverse behaviors than BA-MBRec(GlobalMeanPooling). Furthermore, it facilitates a more personalized and weighted fusion of behavior embeddings derived from various behavior patterns.
- **Effect of Hierarchical Behavior Dependency Modeling:** To isolate the contribution of modeling intra-layer behavioral dependencies enabled by HBDM, we designed the simpler variant, called BA-MBRec(LinearProj). This variant replaces the hierarchical attention mechanism with simple linear layers, which only perform basic semantic space projection without capturing the nuanced similarities between concurrent behaviors at the same hop l. As a result, the significant performance drop of BA-MBRec(LinearProj) compared to the full model validates the necessity of HBDM's specific mechanism for capturing these crucial same-length behavioral dependencies.
- **Effect of Meta Contrastive Network:** We propose another variant, BA-MBRec(UniformCL), which relies solely on contrastive learning to capture mutual information between specific types of behavioral embeddings. Specifically, the MV weight network is removed, and the behavior comparison loss function and the BPR loss function are treated equally during training. The result suggests that by leveraging the meta contrastive network, we can automatically discern the influence between different target-auxiliary behavior pairs, enabling cross-view behavior dependencies to complement each other.
- **Effect of Multi-behavior Contrastive Learning Framework:** We create the model variant BA-MBRec(NoCL) by disabling contrastive learning between target and auxiliary user behaviors. Instead, we solely rely on behavior-aware graph neural networks to capture behavioral relationships. The result suggests that the effectiveness of our model in capturing complex dependencies across different behavior types, while also mitigating the impact of skewed data distribution, and effectively transferring knowledge across behavior views.

5.4 Effect of Context Behaviors (RQ3)

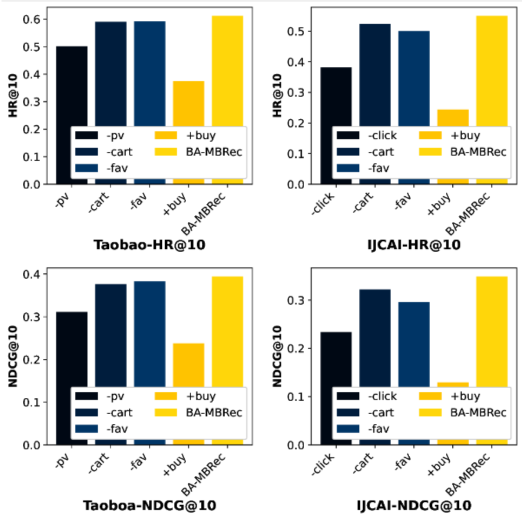

Fig. 2. Impact study of different types of context behavior.

We undertake an additional context experiment to evaluate the effectiveness of integrating various context behaviors into our BA-MBRec framework. We categorize these behaviors according to their respective datasets, and the outcomes are depicted in **Fig. 2**. In this figure, the notations *"-pv"*, *"-cart"*, and *"-fav"* indicate that the model was trained without incorporating the respective viewing, carting, and favoriting behaviors. Conversely, *"+buy"* signifies the variant that solely relies on the target purchase behaviors for predictions. Our findings reveal that our model outperforms all other variants by comprehensively capturing user preferences through context behaviors. Further validation confirms that our model can extract dependency relationships from intricate behavior patterns.

5.5 Performance on Data Sparsity (RQ4)

This section evaluates how our framework performs on user behavior data of varying sparsity, particularly regarding its effectiveness in multiplex behavior recommendation. Adhering to a similar setting as in [20], we generated datasets with varying degrees of sparsity from the Taobao dataset. The results are presented in **Fig. 3.** Performance of BA-MBRec and baseline methods *w.r.t* different data sparsity degrees on Taobao data.. Specifically, users were divided into six groups based on their interaction counts: G_1: (0,4], G_2: (4,5], G_3: (5,6], G_4: (6,7], G_5: (7,8], and G_6: > 8. To be more specific, the right y-axis indicates the indicators evaluated by HR@10 and NDCG@10. The total number of users belonging to each group is shown on the left y-axis of Fig. 3.

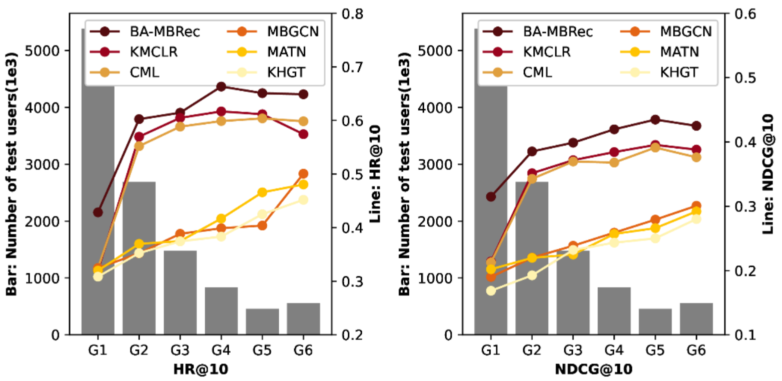

Fig. 3. Performance of BA-MBRec and baseline methods *w.r.t* different data sparsity degrees on Taobao data.

From the result, we could obtain the following findings: (i) Recommendation accuracy across all models improves with increased user interactions, as more data allows for finer-grained behavior modeling. (ii) Contrastive learning models (e.g., KMCLR, CML) outperform those using only auxiliary information (e.g., MBGCN), proving contrastive learning's ability to preserve behavior heterogeneity and semantic diversity via self-supervision. (iii) Our model outperforms other multi-behavior models, even those that proved effective for data sparsity, across all user groups due to its superior ability to extract inter-behavior interaction information.

5.6 Hyperparameter Setting on BA-MBRec (RQ5)

Fig. 4 illustrates the impact of different hyperparameter settings on our framework's performance. Each analysis isolates a single hyperparameter while keeping other parameters at their default values.

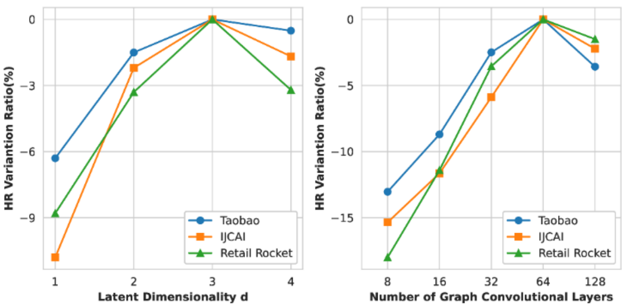

Fig. 4. Hyperparameter analysis of BA-MBRec.

Graph Propagation Layers *l*. Increasing graph convolution layers improved performance up to three layers, suggesting benefits from capturing higher-order information through message passing. However, further stacking introduced noise and over-smoothing, ultimately harming performance [21].

Latent Dimensionality of Embeddings d. Experimenting with embedding dimensions from 8 to 128, we found optimal performance with dimensionality reaching 64. Larger dimensions led to overfitting, yielding marginal gains or performance drops.

5.7 Case Study (RQ6)

This section presents a qualitative study demonstrating our model's interpretability. We visualize the meta-weights to illustrate the contribution of individual context behaviors in predicting the target behavior.

Behavior-specific Contrastive Weight Visualization. We visualize the learned weights $\omega_u^{rt,r}$ for each behavior pair of sampled users. The result is shown in **Fig. 5**, which reflects the customized preferences of individuals. The color bar represents loss weights between behaviors and the target behavior, with darker colors indicating stronger correlation and higher weights. For instance, users "14222" and "19130" are more likely to purchase after clicking than after other actions.

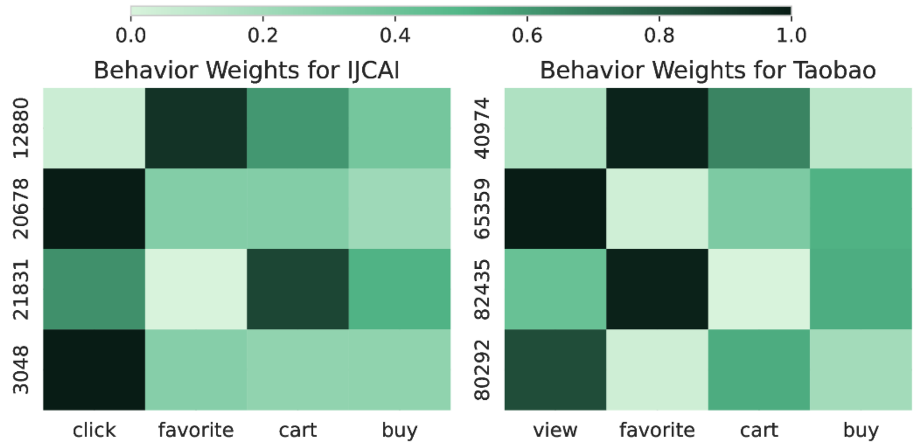

Fig. 5. Behavior-specific contrastive weights learned from data.

6 Conclusion

In this work, we propose a method called BA-MBRec for multi-behavior recommendation tasks. BA-MBRec implicitly integrates behavior pattern construction with hierarchical dependency modeling at each hop and personalized cross-behavior aggregation. Results from experiments on three real-world datasets showcase BA-MBRec's improved performance compared to state-of-the-art baselines. Moreover, ablation studies affirm the effectiveness of its learned feature embeddings.

7 Acknowledgement

This work was supported by the following grants: the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (Grant Nos. 2025C01054, 2025C01212, and 2023C01029), and the Zhejiang Provincial Natural Science Foundation of China (Grant No. LR23F020004). The authors gratefully acknowledge this financial support.

References

1. Wei Y, Wang X, Nie L, et al. Graph-refined convolutional network for multimedia recommendation with implicit feedback[C]//Proceedings of the 28th ACM international conference on multimedia. 2020: 3541-3549.
2. He X, Deng K, Wang X, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation[C]//Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020: 639-648.
3. Wang X, He X, Wang M, et al. Neural graph collaborative filtering[C]//Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. 2019: 165-174.
4. Wei W, Huang C, Xia L, et al. Contrastive meta learning with behavior multiplicity for recommendation[C]//Proceedings of the fifteenth ACM international conference on web search and data mining. 2022: 1120-1128.
5. Shi C, Hu B, Zhao W X, et al. Heterogeneous information network embedding for recommendation[J]. IEEE transactions on knowledge and data engineering, 2018, 31(2): 357-370.
6. Liu Z, Huang C, Yu Y, et al. Fast attributed multiplex heterogeneous network embedding[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. 2020: 995-1004.
7. Yu P, Fu C, Yu Y, et al. Multiplex heterogeneous graph convolutional network[C]//Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2022: 2377-2387.
8. Fu C, Zheng G, Huang C, et al. Multiplex heterogeneous graph neural network with behavior pattern modeling[C]//Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining. 2023: 482-494.
9. Li X, Fu C, Zhao Z, et al. Dual-channel multiplex graph neural networks for recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2025.
10. Xia L, Huang C, Xu Y, et al. Knowledge-enhanced hierarchical graph transformer network for multi-behavior recommendation[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(5): 4486-4493.
11. Yun S, Jeong M, Kim R, et al. Graph transformer networks[J]. Advances in neural information processing systems, 2019, 32.
12. Xia L, Huang C, Xu Y, et al. Multiplex behavioral relation learning for recommendation via memory augmented transformer network[C]//Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 2397-2406.

13. Xia L, Xu Y, Huang C, et al. Graph meta network for multi-behavior recommendation[C]//Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval. 2021: 757-766.
14. Jin B, Gao C, He X, et al. Multi-behavior recommendation with graph convolutional networks[C]//Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval. 2020: 659-668.
15. Wang X, Liu N, Han H, et al. Self-supervised heterogeneous graph neural network with co-contrastive learning[C]//Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining. 2021: 1726-1736.
16. Wu J, Wang X, Feng F, et al. Self-supervised graph learning for recommendation[C]//Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval. 2021: 726-735.
17. Zhu Y, Xu Y, Yu F, et al. Graph contrastive learning with adaptive augmentation[C]//Proceedings of the web conference 2021. 2021: 2069-2080.
18. Xuan H, Liu Y, Li B, et al. Knowledge enhancement for contrastive multi-behavior recommendation [C]//Proceedings of the sixteenth ACM international conference on web search and data mining. 2023: 195-203.
19. Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback[J]. arXiv preprint arXiv:1205.2618, 2012.
20. Park C, Kim D, Han J, et al. Unsupervised attributed multiplex network embedding[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(04): 5371-5378.
21. Luo D, Cheng W, Yu W, et al. Learning to drop: Robust graph neural network via topological denoising[C]//Proceedings of the 14th ACM international conference on web search and data mining. 2021: 779-787.
22. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks[C]//Proceedings of the thirteenth international conference on artificial intelligence and statistics. JMLR Workshop and Conference Proceedings, 2010: 249-256.
23. Loshchilov I, Hutter F. Decoupled weight decay regularization[J]. arXiv preprint arXiv:1711.05101, 2017.
24. Alyafi B, Tushar F I, Toshpulatov Z. Cyclical learning rates for training neural networks with unbalanced data sets[J]. Jmd in medical image analysis and applicationspattern recognition module, 2018.