Neurocomputing 681 (2026) 133379

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Distillation conditional diffusion with spectral-enhanced hierarchical fusion for multi-behavior recommendation

Xiaofei Zhu*, Peng Shan

School of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China

HIGHLIGHTS

- Design a robust residual cascading GCN to capture fine-grained semantic patterns.
- Develop a spectrally-enhanced hierarchical fusion to mitigate embedding fluctuations.
- Experiments on three datasets show the superior performance of the proposed model.

ARTICLE INFO

Communicated by G. Gnecco

Keywords:
Multi-behavior recommendation
Conditional diffusion models
Knowledge distillation
Graph convolutional network

ABSTRACT

Multi-behavior recommendation aims to improve prediction accuracy by leveraging diverse user interactions, with its core challenge lying in effectively modeling the complex relationships among behaviors. Existing methods often struggle to address two key issues: the semantic gap between auxiliary and target behaviors, and the unstable fluctuations within auxiliary behaviors caused by noisy interactions. To this end, we propose DCDRec, a novel framework centered on a Robust Residual Cascading Graph Convolutional Network (RRC-GCN) and a Spectral-enhanced Hierarchical Fusion module (SHF). RRC-GCN suppresses noise in auxiliary behaviors and enhances embedding robustness through target-guided attention masking and distillation conditional diffusion (DCDiff). SHF integrates multi-behavior embeddings via a hierarchical structure to capture inter-behavior correlations and semantic complementarity. Experiments on three real-world datasets show that DCDRec significantly outperforms existing baselines, demonstrating its effectiveness in learning robust and semantically rich representations for multi-behavior recommendation. Our source code is available at https://github.com/SPeng996/DCDRec.

1. Introduction

Multi-behavior recommendation systems (MBR) have emerged as an important research direction and play a crucial role in modern e-commerce and social media platforms [1,7,29,33,34], aiming to improve recommendation accuracy by leveraging users' diverse interaction types. The core challenge of MBR lies in how to effectively model the complex relationships and dependencies among multiple behaviors. To address this challenge, various deep learning-based methods have been proposed. For example, some studies employ Graph Convolutional Networks (GCNs) [2,8,12,13,15] to capture high-order user–item correlations across different behaviors, while others explore Transformer-based architectures [26,29] to effectively exploit multi-behavior interactions for recommendation. Despite their promising performance, these methods often focus on modeling each behavior independently or simply aggregating multi-behavior interactions, which may limit their ability to fully capture the rich and hierarchical relationships across different types of user actions.

In recent years, considerable research efforts have been devoted to modeling the complex relationships among multiple user behaviors. One line of work emphasizes parallel paradigms [17,32], which model diverse behaviors as independent information sources to capture behavior-specific patterns. MB-HGCN [32] constructs a unified graph to capture global user-item relationships, while leveraging specific behavior graphs to model distinct behavioral signals. MuLe [17] further extends this idea by introducing a multi-grained graph learning framework that captures behavioral relations at multiple levels. It emphasizes the complementary nature between auxiliary and target behaviors to better distinguish potential and confirmed user intentions. Another line of work focuses on cascaded paradigms, which attempt to explicitly capture the sequential dependencies among behaviors following the natural

* Corresponding author.
Email addresses: zxf@cqut.edu.cn (X. Zhu), shanp@stu.cqut.edu.cn (P. Shan).

https://doi.org/10.1016/j.neucom.2026.133379
Received 9 January 2026; Received in revised form 18 February 2026; Accepted 14 March 2026
Available online 17 March 2026

cascading order of user actions (e.g., view → cart → buy). CRGCN [31] and MB-CGCN [2] consider the behavior dependency in a behavior chain and leverage a cascading graph convolutional network to learn user and item embeddings. COPF [34] formulates multi-behavior fusion as a combinatorial optimization problem, applying distinct constraints at various stages of each behavior to limit the solution space. CMC-GCN [33] introduces a novel multi-granularity cascading graph convolutional network that captures both fine- and coarse-grained relationships by jointly modeling the user-item interaction graph and its corresponding hypergraph for each behavior.

However, existing approaches still face two critical challenges: **(1) *Semantic disparity between auxiliary and target behaviors.*** Auxiliary and target behaviors often present substantial semantic differences. Most existing methods either treat auxiliary behaviors as supplementary signals to enhance target behavior learning or adopt multi-task learning [30,35] to predict them separately. However, these strategies fail to explicitly model the fine-grained impact of different auxiliary behavior patterns on the target behavior. For instance, users who only view items may have different preferences from those who both view and buy. Although MuLe [17] models the multifaceted relationships between auxiliary and target behaviors to capture richer semantics, it lacks a mechanism to effectively guide target behavior learning from auxiliary behaviors, resulting in limited semantic transfer. **(2) *Unstable fluctuations within auxiliary behaviors*.** Auxiliary behaviors often exhibit unstable fluctuations within their representations, caused by short-term, sporadic, or weakly informative interactions. Existing methods typically fuse auxiliary behavior embeddings using global graph structures or simple attention mechanisms, but they neither suppress these unstable components nor emphasize the stable semantic patterns that are relevant to conversion. Consequently, noisy embedding signals are indiscriminately propagated, obscuring meaningful conversion cues and ultimately degrading recommendation performance.

To address the aforementioned challenges, we propose a novel Distillation Conditional Diffusion with Spectral-enhanced Hierarchical Fusion, called DCDRec, for multi-behavior recommendation. To be specific, DCDRec consists of two key modules, i.e., the Robust Residual Cascading Graph Convolutional Network (RRC-GCN) and the Spectral-enhanced Hierarchical Fusion (SHF). To tackle the semantic disparity among behaviors, RRC-GCN first separates auxiliary interactions into converted behaviors and unconverted behaviors. Converted behaviors refer to interactions that lead to the target behavior (e.g., viewing a product followed by purchasing it), which typically contain informative signals relevant to the target behavior. In contrast, unconverted behaviors are those that do not result in conversion (e.g., viewing a product without a subsequent purchase), and may involve both useful interactions related to the target behavior and noisy interactions irrelevant to it. To mitigate the noisy information present in unconverted behaviors, RRC-GCN introduces a Target-Guided Attention Mask Graph Convolutional Network (TamGCN), which adaptively emphasizes the most informative interactions while suppressing noisy or less relevant ones in auxiliary behaviors. Building upon this, it develops a Distillation Conditional Diffusion (DCDiff) module to enhance the robustness and expressiveness of the learned embeddings for each auxiliary behavior. DCDiff leverages information learned from the denoised unconverted behaviors as conditional signal to guide the diffusion process and then aligns the generated outputs with patterns observed in converted behaviors. To handle the unstable fluctuations within auxiliary behaviors, SHF proposes to integrate multi-behavior embeddings with a spectral-enhanced hierarchical fusion structure, which jointly captures hierarchical correlations, semantic complementarities and spectral dynamics across multi-behavior embeddings. Experimental results on three real-world datasets demonstrate the effectiveness of our proposed approach.

In summary, the main contributions of our work are as follows:

- We propose a novel DCDRec framework to address two key challenges in multi-behavior recommendation, including the semantic gap between auxiliary behaviors and the target behavior, as well as the unstable fluctuations within auxiliary behaviors.
- We design a Robust Residual Cascading Graph Convolutional Network (RRC-GCN) by introducing a Target-Guided Attention Mask Graph Convolutional Network (TamGCN), which adaptively emphasizes informative interactions and suppresses noise within auxiliary behaviors. Furthermore, we develop a Distillation Conditional Diffusion (DCDiff) module that utilizes the denoised unconverted behaviors as a conditional signal to guide the diffusion process, as well as to align the generated outputs with the converted behaviors.
- We propose a Spectral-enhanced Hierarchical Fusion (SHF) module to handle the unstable fluctuations within auxiliary behaviors. This module employs a spectral-enhanced hierarchical fusion architecture to jointly capture the hierarchical correlations, semantic complementarities, and spectral dynamics among multi-behavior embeddings.
- We conduct a comprehensive experimental evaluation of our proposed approach on three datasets, including Taobao, Tmall and Jdata. The proposed approach yields relative improvements of up to 5.44% in H@10 and 9.19% in N@10 over the most competitive baselines across these datasets.

2. Related work

2.1. Multi-behavior recommendation

Multi-behavior recommendation methods aim to leverage users' diverse interaction data to alleviate the data sparsity problem in target behaviors and have attracted widespread attention in recent years. Early multi-behavior recommendation methods extend traditional matrix factorization approaches to accommodate multi-behavior data [16,24,25]. Moreover, some methods develop new sampling strategies using auxiliary behavioral signals [4,6], but the rich behavioral information are not fully explored.

With the rise of deep learning, Deep Neural Networks (DNN) and Graph Convolutional Networks (GCN) have been widely applied in the recommendation domain [2,3,12,18], as they can better capture the multifaceted information across different behaviors. DNN-based methods typically leverage neural networks to extract information from user–item interactions and embed this information into representations. For example, DIPN [7] and MATN [29] adopt attention mechanisms to capture implicit relationships among behaviors. However, most DNN-based models fail to capture high-order information, resulting in suboptimal performance. In contrast, GCN-based models can capture high-order relationships between users and items, making them the mainstream recommendation paradigm. RGCN [23], GNMR [28], and MBGCN [12] represent multi-behavior interactions on a unified graph and further aggregate behavior embeddings via GCN.

Some efforts consider the hierarchical correlations among behaviors, assuming that user behaviors exhibit a progressive relationship. NMTR [5], CRGCN [31], and MB-CGCN [2] leverage a cascading framework to learn the representation of each behavior and employ a multi-task learning (MTL) module for joint optimization. However, these methods offer a limited perspective on multi-behavior fusion, which hinders their ability to accurately capture diverse user behavior patterns during the fusion stage. COPF [34] introduces a combinatorial optimization approach to model various patterns of user behavior and effectively fuse multiple behavioral features. CMC-GCN [33] utilizes a multi-granularity cascading graph convolutional network on both the user-item interaction graph and its corresponding hypergraph to simultaneously learn fine- and coarse-grained patterns for each behavior. However, while these cascading approaches effectively capture sequential dependencies, they

often struggle to bridge the semantic disparity between auxiliary and target behaviors or adaptively suppress inherent noise in non-conversion signals. To address the imbalance in behavior interactions, recent parallel approaches aim to learn embeddings separately on each behavior graph. MB-HGCN [32] initializes GCN with global embeddings from a unified graph and employs a MTL module to effectively refine behavior embeddings for performance improvement. MuLe [17] proposes a multi-grained graph learning strategy to model the multifaceted relationships between auxiliary and target behaviors from multiple aspects.

2.2. Diffusion-based recommendation

In recent years, diffusion models have emerged as a powerful generative modeling paradigm. They provide a novel generative perspective, enabling the modeling of high-dimensional and complex user preference distributions, and have become a promising direction in recommender systems. DiffRec [27] and DiffKG [11] leverage diffusion models to denoise user interactions and knowledge graphs, thereby reducing structural noise and improving recommendation performance considerably. DiffKG [11] learns collaborative signals from user-item interactions and utilizes them to guide the denoising process of knowledge graph diffusion. DiffGraph [19] and RecDiff [20] enhance graph-based recommendations by applying diffusion mechanisms directly to GCN-derived node embeddings. This process smooths representations, captures high-order structural relations, and mitigates noise, ultimately improving both representation quality and recommendation performance.

Our method extends existing paradigm of diffusion models in two-folds. First, we leverage the unconverted behaviors as condition to preliminarily filter out irrelevant information from the auxiliary behavior, retaining only those elements of user intent that positively contribute to conversions towards the target behavior. Second, the resulting denoised auxiliary embedding of the reverse process is then distilled towards the converted behavior rather than reconstructing the original input as in existing traditional diffusion models.

This model design enables our approach to effectively utilize both converted and unconverted behaviors to learn high-quality auxiliary behavior representations. It achieves this by reducing the noise and irrelevant information in the target behavior from different perspectives.

3. Preliminaries

Problem definition. Let $\mathcal{U} = \{u_1, u_2, \dots, u_M\}$ and $\mathcal{I} = \{i_1, i_2, \dots, i_N\}$ denote the sets of users and items, where M and N represent the number of users and items, respectively. $\mathcal{B} = \{view, \dots, buy\}$ is the set of behaviors, and h denotes the target behavior (i.e., *buy*). A user-item graph on behavior $b \in \mathcal{B}$ is indicated by $\mathcal{G}_b = (\mathcal{V}, \mathcal{E}_b)$ where $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$, and $\mathcal{E}_b$ denotes the set of edges corresponding to behavior b. $R_b \in \mathbb{R}^{M \times N}$ is the user-item interaction matrix where $R_b(u,i)$ is 1 if $(u,i) \in \mathcal{E}_b$; otherwise, it is 0. Let $A_b \in \mathbb{R}^{(M+N)\times(M+N)}$ and $\tilde{A}_b \in \mathbb{R}^{(M+N)\times(M+N)}$ denote the adjacency matrix of $\mathcal{G}_b$ and its corresponding symmetric normalized matrix, respectively, which are defined as follows:

$$A_b = \begin{bmatrix} 0 & R_b \\ R_b^\top & 0 \end{bmatrix}, \tag{1}$$

$$\tilde{A}_b = D^{-\frac{1}{2}} A_b D^{-\frac{1}{2}}, \tag{2}$$

where D is a diagonal degree matrix.

The objective of multi-behavior recommendation is to model the user set $\mathcal{U}$ and the item set $\mathcal{I}$, along with the set of multi-behavior interactions $\mathcal{E}$ as input. The goal is to produce a ranking score $y_h(u,i)$ between user u and item i, indicating the probability that user u will interact with item i in the target behavior h.

LightGCN. Given the effectiveness of LightGCN in graph-based embedding learning, we adopt a variant of LightGCN with normalization [17,32] as the graph encoder to learn user and item embeddings. The LightGCN with L layers is defined as follows:

$$E = LightGCN(A, E^0) = E^0 + \sum_{l=1}^{L} \frac{E^l}{l}, \tag{3}$$

where $E^0 \in \mathbb{R}^{(M+N)\times d}$ is the initial embedding matrix and $E^l \in \mathbb{R}^{(M+N)\times d}$ denotes the embedding matrix obtained at the l-th layer. Then, a row-wise L_2-normalization operation is applied on the user-item interaction graph defined as $E^l = normalize(\tilde{A}E^{l-1})$.

4. Method

In this section, we introduce the Distillation Conditional Diffusion with Spectral-enhanced Hierarchical Fusion for multi-behavior recommendation, which consists of two parts: (1) Robust Residual Cascading Graph Convolutional Network (RRC-GCN); (2) Spectral-enhanced Hierarchical Fusion Module (SHF). Fig. 1 illustrates the overall architecture of the proposed framework.

4.1. Robust residual cascading graph convolutional network

In multi-behavior recommendation, users often interact with items through different types of behavior (e.g., view, cart, buy), with each carrying distinct semantic meanings and conversion implications. To effectively model such diverse and interrelated behaviors, we design a robust residual cascading graph convolutional network.

4.1.1. Residual cascading graph learning

Following existing deep learning-based recommendation methods [17,32,34], we randomly initialize embeddings of users $\mathcal{U}$ and items $\mathcal{I}$ as $E_{init}(u) \in \mathbb{R}^{M\times d}$ and $E_{init}(i) \in \mathbb{R}^{N\times d}$, where d denotes the embedding size, and $E_{init} \in \mathbb{R}^{(M+N)\times d}$ denotes the initial learnable embedding matrix. We first construct a global interaction graph, which integrates all user-item interaction behaviors. Let $\mathcal{G}_{global} = (\mathcal{V}, \mathcal{E}_{global})$ denote the global interaction graph, where $\mathcal{E}_{global} = \cup_{b\in\mathcal{B}}\mathcal{E}_b$ is the integrated interactions of all behaviors. We employ LightGCN to learn the user and item embeddings in the global interaction graph E_{global} as follows:

$$E_{global} = LightGCN(A_{global}, E_{init}), \tag{4}$$

where A_{global} is the adjacency matrix of $\mathcal{G}_{global}$.

After obtaining the representations from the global user–item interaction graph, we further learn fine-grained representations for each behavior-specific graph $\mathcal{G}_b$. Specifically, for a behavior sequence $\mathcal{B}_{seq} = \{b_1, b_2, \dots, b_S\}$ with S behaviors, we initialize the embeddings of the first behavior using the global embeddings, and apply LightGCN to learn its embeddings as follows:

$$E_{b_1} = LightGCN(A_{b_1}, E_{global}). \tag{5}$$

For each subsequent behavior $b_s (s > 1)$, we initialize its embeddings by adaptively fusing the global embeddings and all learned embeddings of previous behaviors.

$$\hat{E}_{b_s} = w^0_{b_s} E_{global} + \sum_{j=1}^{s-1} w^j_{b_s} E_{b_j}, \tag{6}$$

where the weights $[w^0_{b_s}, w^1_{b_s}, \dots, w^{s-1}_{b_s}]$ are learnable parameters that adaptively control the contribution from each source. Finally, the embedding for the behavior b_s is learned by LightGCN on its behavior-specific graph:

$$E_{b_s} = LightGCN(A_{b_s}, \hat{E}_{b_s}). \tag{7}$$

Through this process, we obtain behavior-specific embeddings $\{E_{view}, E_{cart}, \dots, E_{buy}\}$.

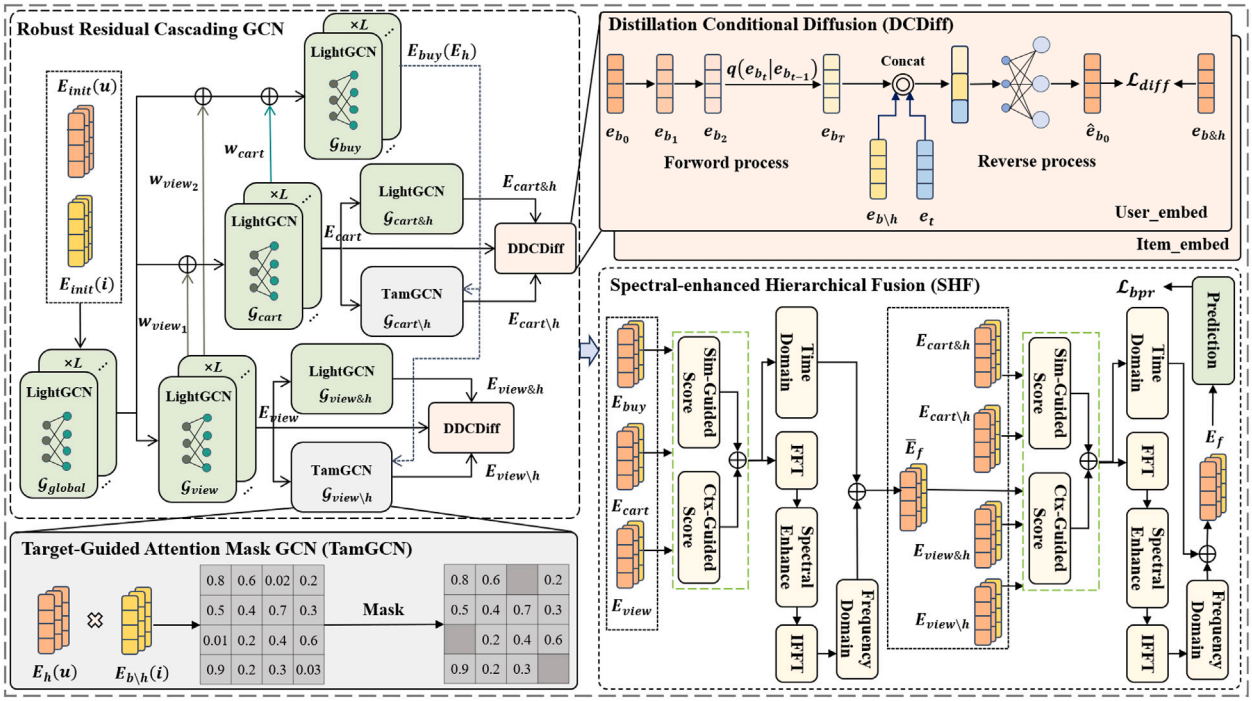

Fig. 1. The overall architecture of DCDRec. We utilize three behaviors (i.e., view, cart and buy) as an example, wherein buy is the target behavior.

4.1.2. Target-guided attention mask GCN (TamGCN)

Let $\mathcal{B}_{aux}$ denote the set of auxiliary behaviors, i.e., all behaviors except the target behavior h, then $\mathcal{B}_{con} = \{b\&h | b \in \mathcal{B}_{aux}\}$ denotes the set of converted behaviors, representing that each auxiliary behavior $b \in \mathcal{B}_{aux}$ eventually leads to the target behavior. Similarly, $\mathcal{B}_{ucon} = \{b\backslash h | b \in \mathcal{B}_{aux}\}$ denotes the set of unconverted behaviors, where the auxiliary interaction does not result in the target behavior.

We separately construct interaction graphs for these two sets. For each converted behavior graph $\mathcal{G}_{b\&h} = (\mathcal{V}, \mathcal{E}_{b\&h})$, the converted behavior embedding $E_{b\&h}$ is obtained via LightGCN:

$$E_{b\&h} = LightGCN(A_{b\&h}, E_b), \tag{8}$$

where $A_{b\&h}$ is the adjacency matrix of $\mathcal{G}_{b\&h}$, and E_b is the embedding of the original auxiliary behavior.

To minimize noise and filter out interactions irrelevant to the target behavior on the unconverted behavior graph $\mathcal{G}_{b\backslash h} = (\mathcal{V}, \mathcal{E}_{b\backslash h})$, we introduce the Target-Guided Attention Mask GCN (TamGCN). This module processes the unconverted behavior graph by utilizing the target behavior embedding E_h as a supervisory signal for the attention mechanism.

Over L layers, TamGCN derives the final unconverted behavior representation $E_{b\backslash h}$ through a multi-layer aggregation strategy:

$$\begin{aligned} E_{b\backslash h} &= TamGCN(\mathcal{G}_{b\backslash h}, E_h, E^0_{b\backslash h}) \\ &= E^0_{b\backslash h} + \sum_{l=1}^{L} \frac{E^l_{b\backslash h}}{l}, \end{aligned} \tag{9}$$

where $E^0_{b\backslash h}$ is initialized with E_b. At each layer l, the embedding $E^l_{b\backslash h}$ is refined via an attention-based neighborhood aggregation:

$$E^l_{b\backslash h} = normalize(A^l_{b\backslash h}, E^{l-1}_{b\backslash h}), \tag{10}$$

where $A^l_{b\backslash h} \in \mathbb{R}^{(M+N)\times(M+N)}$ represents the dynamic attention-weighted adjacency matrix. For any interaction $(u, i) \in \mathcal{E}_{b\backslash h}$, the specific weight assigned to the connection between user u and item i is calculated as:

$$A^l_{b\backslash h}(u, i) = \frac{\exp(a_{b\backslash h}(u, i))}{\sum_{j \in \mathcal{R}_{b\backslash h}(u)} \exp(a_{b\backslash h}(u, j))}, \tag{11}$$

where $\mathcal{R}_{b\backslash h}(u)$ denotes the neighborhood of u within $\mathcal{G}_{b\backslash h}$. The underlying attention score $a_{b\backslash h}(u, i)$ is determined by the alignment between the target embedding and the current item features, defined as $a_{b\backslash h}(u, i) = E_h(u) \cdot E^{l-1}_{b\backslash h}(i)$.

After computing the normalized attention weights $A^l_{b\backslash h}(u, i)$ for all $(u, i) \in \mathcal{E}_{b\backslash h}$, we filter out low-relevance connections by defining the masked attention matrix as:

$$A^l_{b\backslash h}(u, i) = \begin{cases} A^l_{b\backslash h}(u, i), & \text{if } A^l_{b\backslash h}(u, i) \geq \tau \\ 0, & \text{otherwise} \end{cases}, \tag{12}$$

where τ is a threshold derived from a pre-defined quantile of the global attention weights. This masking operation effectively suppresses interactions with low semantic relevance to the target behavior, enhancing the robustness of the learned embeddings.

The attention weights from items to users, denoted as $A^l_{b\backslash h}(i, u)$, are computed in a symmetric manner by utilizing $E_h(i)$ and $E^{l-1}_{b\backslash h}(u)$. For any interaction $(u, i) \notin \mathcal{E}_{b\backslash h}$, we set $A^l_{b\backslash h}(u, i) = 0$ and $A^l_{b\backslash h}(i, u) = 0$. This ensures that the attention mechanism strictly adheres to the topological structure of $\mathcal{G}_{b\backslash h}$, thereby preserving the sparsity and inherent structural constraints of the unconverted behavior graph.

4.1.3. Distillation conditional diffusion

To mitigate the semantic disparity between auxiliary and target behaviors in multi-behavior recommendation, we propose the Distillation Conditional Diffusion Module (DCDiff). Rather than directly optimizing auxiliary embeddings independently, DCDiff treats them as input and employs the unconverted behavior embeddings as conditional signals in the diffusion process. These diffused embeddings are then aligned with the converted behavior embeddings based on a distillation process, thereby guiding auxiliary embeddings to better reflect the patterns and preferences captured in the target behaviors.

Forward diffusion. For each user (item) embedding ${e_b}^u$ (${e_b}^i$) $\in E_b$, Gaussian noise is incrementally added to the input embeddings over T steps. For simplicity and clarity, we omit the superscript (i and u) when it is unambiguous. The embedding at the t-th step is denoted as e_{b_t}, with the 0-th step being the input embeddings, i.e., $e_{b_0} = e_b$. The embedding at step t is calculated from the $(t-1)$-th step embedding as follows:

$$q(e_{b_t}|e_{b_{t-1}}) = \mathcal{N}(e_{b_t}; \sqrt{1-\beta_t}e_{b_{t-1}}, \beta_t I), \tag{13}$$

where β_t controls the noise scale and $\mathcal{N}$ denotes the Gaussian distribution. As $t \to T$, the embedding e_{b_t} converges to pure Gaussian noise. To avoid recursive computation, we follow the standard formulation of the diffusion process:

$$\alpha_t = 1 - \beta_t, \quad \bar{\alpha}_t = \prod_{t'=1}^{t} \alpha_{t'}, \tag{14}$$

$$\begin{aligned} e_{b_t} &= \sqrt{\alpha_t} e_{b_{t-1}} + \sqrt{1-\alpha_t}\xi_1 \\ &\Rightarrow \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}\, e_{b_{t-2}} + \sqrt{1-\alpha_{t-1}}\,\xi_2\right) + \sqrt{1-\alpha_t}\,\xi_1 \\ &\Rightarrow \sqrt{\bar{\alpha}_t}\, e_{b_0} + \sqrt{1-\bar{\alpha}_t}\,\xi_t', \quad \xi \mapsto \mathcal{N}(0, I), \end{aligned} \tag{15}$$

where ξ_* denotes noise vector injected into e_b at each step.

Reverse diffusion. The reverse diffusion process aims to iteratively denoise the embedding e_{b_T} to recover its clean initial representation e_{b_0}. We first introduce a conditional reverse diffusion mechanism guided by the unconverted behavior signals. The rationale behind this is that the user intent contained in the unconverted behavior is generally more ambiguous, and we utilize the unconverted behavior information to preliminarily filter out irrelevant information for the auxiliary behavior, retaining only those aspects of user intent that positively contribute to conversions towards the target behavior. Specifically, the unconverted behavior embedding $e_{b\setminus h}$ learned via TamGCN serves as the conditional signal to guide the denoising process, attempting to alleviate semantically target-irrelevant information in the auxiliary behavior. Formally, the reverse process at each step t is defined as:

$$p_\theta(e_{b_{t-1}} \mid e_{b_t}) = \mathcal{N}\left(e_{b_{t-1}}; \mu_\theta(e_{b_t}, e_{b\setminus h}, t), \Sigma_\theta(e_{b_t}, e_{b\setminus h}, t)\right), \tag{16}$$

where $\mu_\theta(e_{b_t}, e_{b\setminus h}, t)$ and $\Sigma_\theta(e_{b_t}, e_{b\setminus h}, t)$ are the mean and covariance of the Gaussian distribution predicted by a neural network with learning parameter θ.

Behavior distillation. The resulting denoised auxiliary embedding $\hat{e}_{b_0}$ is then distilled towards the converted behavior embedding $e_{b\&h}$, enabling the auxiliary representation to align with the target behavior while maintaining robustness. It is worth noting that, unlike traditional diffusion models which are designed to reconstruct the original input, we introduce a distillation process to alter the objective of the reverse denoising process to align with the embeddings of the converted behavior. By treating the converted behavior embeddings, which encode true user preferences, as the reconstruction objective, our diffusion model can learn to distill the most relevant preference information for the target behavior.

Optimization. To optimize the proposed Distillation Conditional Diffusion (DCDiff) module, we define the evidence lower bound (ELBO) using the converted behavior embeddings $e_{b\&h}$:

$$\begin{aligned} log\ p(e_{b\&h}) \geq\ & \underbrace{\mathbb{E}_{q(e_{b_1}|e_{b_0})}\left[log\ p_\theta(e_{b\&h}|e_{b_1})\right]}_{\text{(distillation term)}} \\ & - \underbrace{\sum_{t=2}^{T} \mathbb{E}_{q(e_{b_t}|e_{b_0})}\left[D_{\mathrm{KL}}\left(q(e_{b_{t-1}}|e_{b_t}, e_{b_0}) || p_\theta(e_{b_{t-1}}|e_{b_t}, e_{b\setminus h})\right)\right]}_{\text{(denoising matching term)}}, \end{aligned} \tag{17}$$

where the first term represents the distillation term (Note that this is different from the reconstruction term in traditional diffusion model), and the second term indicates the denoising matching term. The objective of the second term minimizes the KL divergence between the true distribution $q(e_{b_{t-1}}|e_{b_t}, e_{b_0})$ and the learned reverse transition $p_\theta(e_{b_{t-1}}|e_{b_t}, e_{b\setminus h})$. Similar to previous works [10,27], we ignore the learning of $\Sigma_\theta(e_{b_t}, t)$ and set $\Sigma_\theta(e_{b_t}, t) = \sigma^2(t)I = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t I$. The $\mathcal{L}_t$ for denoising matching term can thus be written as:

$$\mathcal{L}_t = \mathbb{E}_{q(e_{b_t}|e_{b_0})}\left[\frac{1}{2}\left(\frac{\bar{\alpha}_{t-1}}{1-\bar{\alpha}_{t-1}} - \frac{\bar{\alpha}_t}{1-\bar{\alpha}_t}\right)\left\|\hat{e}_\theta(e_{b_t}, e_{b\setminus h}, t) - e_{b\&h}\right\|_2^2\right], \tag{18}$$

where $\hat{e}_\theta(e_{b_t}, e_{b\setminus h}, t)$ denotes the embedding e_{b_0} predicted by e_{b_t}, $e_{b\setminus h}$ and t.

To explicitly transfer knowledge from the converted behavior to the auxiliary behavior, we treat the reconstruction term at $t = 1$ as a distillation process. To be specific, the converted behavior embedding $e_{b\&h}$ serves as the teacher, and the denoised auxiliary embedding predicted by the reverse diffusion process serves as the student. The squared loss between them attempts to align the auxiliary embedding with the target behavior embedding:

$$\mathcal{L}_1 = \mathbb{E}_{q(e_{b_t}|e_{b_0})}\left[\left\|\hat{e}_\theta(e_{b_1}, e_{b\setminus h}, 1) - e_{b\&h}\right\|_2^2\right]. \tag{19}$$

To determine the diffusion step t, we adopt a uniform sampling strategy. Formally, the diffusion loss $\mathcal{L}_{diff}$ is defined as:

$$\mathcal{L}_{diff} = \mathbb{E}_{t\sim\mathbf{U}(1,t)}\mathcal{L}_t. \tag{20}$$

4.2. Spectral-enhanced hierarchical fusion (SHF)

After obtaining embeddings from the previous modules, we introduce the Spectral-enhanced Hierarchical Fusion (SHF) module, which jointly captures hierarchical correlations, semantic complementarities, and spectral dynamics across embeddings. SHF first performs fusion of specific behavior embeddings. Specifically, it learns weights for embeddings of each specific behavior through two complementary scoring mechanisms, i.e., the Ctx-Guided and the Sim-Guided scores. The former models the contextual interaction between the target behavior and each specific behavior via a learnable linear projection based on their concatenation, and the latter directly measures the semantic similarity between each specific behavior embedding and the target embedding.

Formally, given the target behavior embeddings E_h and a specific behavior embedding E_b ($b \in \mathcal{B}_{seq}$), the Ctx-Guided score is calculated as:

$$c_b = \mathbf{W}(E_h || E_b) + \mathbf{b}, \tag{21}$$

where $||$ denotes the concatenation operation. Semantic relevance is measured as:

$$s_b = sim(E_h, E_b) = \frac{E_h \cdot E_b}{\|E_h\|\,\|E_b\|}. \tag{22}$$

The Sim-Guided score is then integrated with the Ctx-Guided score through a learnable parameter α, and normalized via a softmax to obtain the attention weights:

$$\tilde{c}_b = c_b + \alpha \cdot s_b, \tag{23}$$

$$\bar{c}_b = \frac{\exp(\tilde{c}_b)}{\sum_{b\in\mathcal{B}_{seq}} \exp(\tilde{c}_b)}. \tag{24}$$

We aggregate the behavior embeddings weighted by $\bar{c}_b$ to obtain the time-domain intermediate embedding $\bar{E}_z^t$:

$$\bar{E}_z^t = \sum_{b\in\mathcal{B}_{sg}} \bar{c}_b E_b. \tag{25}$$

To capture the frequency characteristics, we perform a real-valued Fast Fourier Transform (FFT) [21] over the specific behavior embeddings E_b. A learnable frequency filter $\Gamma \in \mathbb{R}^{d/2+1}$ is applied to modulate

the amplitude of each frequency component. Finally, an inverse FFT maps the filtered spectrum back to the time domain and the result is further aggregated using the same attention weights:

$$\bar{E}_z^f = \sum_{b \in \mathcal{B}_{sg}} \bar{c}_b \left(\mathcal{F}^{-1}\left(\Gamma \odot \mathcal{F}(E_b)\right)\right). \tag{26}$$

where $\odot$ denotes the element-wise multiplication with broadcasting, $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the FFT and inverse FFT operations, respectively. This allows the model to emphasize informative spectral bands while suppressing noisy or redundant frequencies.

Finally, the time domain and frequency domain enhanced representations are aggregated via a balancing parameter β to produce the module output:

$$\bar{E}_z = \beta \bar{E}_z^t + (1-\beta)\bar{E}_z^f. \tag{27}$$

After completing the specific behaviors fusion, SHF further fuses $\bar{E}_z$ with the converted and unconverted behavior embeddings $E_{b'}$ ($b' \in \mathcal{B}', \mathcal{B}' = \mathcal{B}_{con} \cup \mathcal{B}_{ucon}$) using the same fusion mechanism. Note that each stage leverages independent linear layers and frequency filters, with each linear layer has a weight matrix $\mathbf{W} \in 2d \times 1$ and a bias term $\mathbf{b} \in \mathbb{R}$. The final fused representation is denoted as E_z. SHF outputs the final fused embeddings $E_z(u)$ and $E_z(i)$ for all users and items.

4.3. Optimization objective and training loss

With the learned embeddings E_z on hand, the prediction score $\hat{y}(u,i)$ between user u and item i is measured as:

$$\hat{y}(u,i) = E_z(u) \cdot E_z(i). \tag{28}$$

Following previous studies, we apply the Bayesian personalized ranking (BPR) loss [22]. This objective encourages the model to assign higher prediction values to observed interactions compared to unobserved ones. Formally, let $\mathcal{E}_h$ denote the set of interactions under the target behavior; the loss function $\mathcal{L}_{bpr}$ is defined as:

$$\mathcal{L}_{bpr} = \sum_{(u,i,j) \in O_h} -ln\,\sigma\left(\hat{y}(u,i) - \hat{y}(u,j)\right), \tag{29}$$

where $\sigma(\cdot)$ is the sigmoid function, O_h is the set of triplets (u,i,j) where $(u,i) \in \mathcal{E}_h$ represents a positive sample, and $(u,j) \notin \mathcal{E}_h$ is a negative sample which is randomly selected. The total loss function of DCDRec is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{bpr} + \lambda \mathcal{L}_{diff} + \gamma \left\|\Theta\right\|_2^2. \tag{30}$$

where Θ represents the set of all model parameters, λ and γ control the weight of the diffusion loss and the parameter regularization term, respectively.

4.4. Inference process

During the inference phase, DCDRec generates recommendations via a streamlined forward pass. Given a user u, a candidate item i, and their historical multi-behavior interaction graphs $\mathcal{G}$, the model first extracts behavior-specific embeddings through the RRC-GCN module. To handle noisy auxiliary signals, the DCDiff module performs a fast T-step reverse denoising process, utilizing unconverted behaviors as conditions to refine the embeddings toward converted behavior representations. Finally, the SHF module integrates these multi-behavior embeddings in the time and frequency domain, and the final recommendation score is calculated via the inner product of the fused user and item representations, as detailed in Algorithm 1.

Algorithm 1: Inference procedure of DCDRec.

Input: User u, candidate item i, historical multi-behavior graphs $\mathcal{G} = \{\mathcal{G}_{view}, \mathcal{G}_{cart}, \mathcal{G}_{view\&h}, \mathcal{G}_{view\setminus h}, \dots, \mathcal{G}_{buy}\}$, initial embeddings E_{init}.

Output: Recommendation score $\hat{y}(u,i)$.

```
// Step 1: Multi-behavior Feature Extraction
```

$E_{global} \leftarrow$ LightGCN($\mathcal{G}, E_{init}$);
$E_{view}, E_{cart}, E_{buy} \leftarrow$ LightGCN($\mathcal{G}, E_{global}$) using Eqs. (5)–(7);

```
// Step 2: Feature Denoising and Alignment
```

Extract unconverted behavior signals $E_{view\setminus h}, E_{cart\setminus h}$ via TamGCN;

for $t = T$ **to** 1 **do**

  Refine auxiliary embeddings $E^{(t-1)}_{view\&h}, E^{(t-1)}_{cart\&h}$ via Reverse Diffusion process (DCDiff) conditioned on $E_{\cdot\setminus h}$ using Eq. (16);

```
// Step 3: Spectral-enhanced Hierarchical Fusion
```

$E_z(u), E_z(i) \leftarrow$ SHF($\cdot$) using Eqs. (21)–(27);

```
// Step 4: Score Prediction
```

Calculate the final preference score using Eq. (28);

return $\hat{y}(u,i)$;

5. Experiments

5.1. Experimental settings

5.1.1. Datasets

Our experiments were conducted on three publicly available real-world benchmark datasets, including Taobao, Tmall and Jdata. The detailed statistics of each dataset are shown in Table 1, the percentage of each auxiliary behavior quantifies the conversion rate from the auxiliary behavior to the target behavior (e.g., view → buy).

Taobao[1] is derived from real-world interaction logs on Taobao, one of the most widely used online shopping platforms in China. It contains 15,449 users and 11,953 items with three types of user behaviors, i.e., view, cart and buy.

Tmall[2] is collected from Tmall, a major Chinese e-commerce platform. This dataset contains 41,738 users and 11,953 items, including four types of user behaviors: $view, collect, cart$ and buy.

Jdata[3] is collected from JD, one of the largest e-commerce platforms in China. It contains interaction records from 93,334 users and 24,624 items, involving the same types of user behaviors as the Tmall dataset.

5.1.2. Evaluation metrics

Following [2,17,31,33], we adopt leave-one-out strategy: (1) the test set consists of the last interacted item and all uninteracted items for each user, (2) the second most recently interacted item for each user forms the validation set for hyperparameter tuning. In order to evaluate the performance of our model, we adopt Hit Ratio (HR@K) and Normalized Discounted Cumulative Gain (NDCG@K) as evaluation metrics. We report the average metrics of all users in the test set with K set to 10 and 20.

5.1.3. Baselines

We evaluated the performance of the DDCDRec framework by comparing it against two categories of recommendation methods.

- **Single-behavior recommendation baselines**:

[1] https://www.taobao.com/
[2] https://www.tmall.com/
[3] https://www.jd.com/

Table 1
Statistics of the experimental datasets.

Dataset	Users	Items	Views	Collects	Carts	Buys
Taobao	15,449	11,953	873,954 (9%)	–	195,476 (10%)	92,180
Tmall	41,738	11,953	1,813,498 (12%)	221,514 (12%)	1996 (15%)	255,586
Jdata	93,334	24,624	1,681,430 (16%)	45,613 (43%)	49,891 (57%)	321,883

  - **MF-BPR** [22] assumes that the predicted scores of positive samples are higher than those of negative samples, optimizing recommendation systems via Bayesian Personalized Ranking (BPR) loss.
  - **NeuMF** [9] combines a shallow generalized matrix factorization model with a deep multilayer perceptron to model user-item interactions.
  - **LightGCN** [8] leverages higher-order connections in the user-item bipartite graph, simplifying the GCN architecture by retaining only the core neighbor aggregation.
- **Multi-behavior recommendation baselines**:
  - **RGCN** [23] designs separate propagation layers for different relations, effectively modeling heterogeneous edges in multi-behavior recommendation.
  - **GNMR** [28] propagates embeddings on a unified multi-behavior graph, using a relation aggregation network to capture dependencies and heterogeneity.
  - **NMTR** [5] employs a cascading neural network to predict interaction scores for each behavior and passes them sequentially, jointly optimized via multi-task learning.
  - **MBGCN** [12] learns user preferences on a unified interaction graph, modeling the impact of multiple behaviors on the target behavior, and introduces an item-item graph for richer interaction information.
  - **CRGCN** [31] uses a cascaded residual network to explore connections between different behaviors via embedding propagation and multi-task learning.
  - **MB-CGCN** [2] models behavior dependencies in a behavior chain with a cascading GCN, transforming previous behavior features as input to the next.
  - **MB-HGCN** [32] initializes GCNs on individual graphs using global embeddings from a unified graph, enabling multi-task learning to leverage refined behavior embeddings.
  - **MuLe** [17] designs a multi-grained graph learning strategy to capture diverse behavior aspects, from unified to behavior-specific interactions.
  - **COPF** [34] treats multi-behavior fusion as a combinatorial optimization problem, imposing constraints at various stages to restrict the solution space and enhance fusion efficiency.
  - **CMC-GCN** [33] uses a multi-granularity cascading GCN to capture fine- and coarse-grained user-item relations via behavior-specific graphs and hypergraphs.

5.1.4. Implementation details

The implementation of our model is based on PyTorch. To ensure a fair comparison, we set the embedding size to 64, the batch size to 1024, and the number of epochs to 200, following previous methods. We utilize the Adam optimizer [14], and employ grid search to tune the learning rate and regularization coefficient in the $\{1e^{-3}, 1e^{-4}, 5e^{-4}, 1e^{-5}\}$ and $\{0, 1e^{-3}, 1e^{-4}, 1e^{-5}\}$ ranges. For the number of GCN layers in each behavior, we consider options for $L_{LightGCN}$ and L_{TamGCN} among $\{1, 2, 3\}$ and $\{1, 2, 3, 4, 5\}$, respectively. The coefficient λ is adjusted within $\{0.05, 0.1, 0.3, 0.5, 0.8\}$ to investigate its impact on model performance. Additionally, the quantile-based threshold τ is tuned across $\{0.4, 0.5, 0.6, 0.7, 0.8\}$ to control the pruning intensity of the unconverted behavior graph. For other methods, we followed the hyperparameter ranges described in the corresponding papers, and an early stopping strategy is adopted during the training stage. We repeated each experiment 5 times with different random seeds and reported the average test performance.

5.2. Overall performance

The overall performance of our proposed DCDRec model and other baseline methods on all three datasets is reported in Table 2. We derive the following key findings from the results. First, among single-behavior baselines, LightGCN achieves the best performance on all datasets, outperforming both MF-BPR and NeuMF. This superiority is mainly attributed to its ability to capture high-order collaborative signals through graph-based propagation. Second, multi-behavior recommendation methods are generally superior to single-behavior approaches across all evaluation metrics and datasets. This observation confirms that modeling multiple behaviors can provide complementary information and more accurately capture user preferences. Third, among multi-behavior recommendation methods, MuLe achieves substantially better performance than other baselines. A key reason is that MuLe explicitly models target-related behaviors, which provide more direct supervision for learning target behavior representations. In contrast, other multi-behavior methods, including cascading approaches (e.g., COPF and CMC-GCN) and parallel approaches such as MB-HGCN, mainly rely on generic auxiliary behaviors and do not explicitly distinguish whether an interaction contributes to the target behavior.

Finally, our proposed DCDRec consistently achieves the best overall performance among all competitors. Compared with the strongest baseline, DCDRec improves HR@10 and NDCG@10 by 5.23% and 9.19% on the Taobao dataset, 5.44% and 8.38% on the Tmall dataset, and 1.28% and 2.20% on the JData dataset, respectively. The superior performance confirms that our proposed approach can effectively suppress noise within auxiliary behaviors as well as mitigate the semantic gap between these auxiliary behaviors and the target behavior.

5.3. Ablation study

To further validate the effectiveness of each key component in DCDRec, we conducted ablation studies with seven variants:

- w/o Cond: We perform the diffusion process of auxiliary embeddings without considering the unconverted behavior as a conditional signal.
- w/o Dist: We remove the distillation mechanism and utilize the original auxiliary behavior representation as the reconstruction target rather than aligning it with the converted behavior representation.
- w/o Diff: We replace the iterative diffusion denoising process with a direct knowledge distillation objective.
- w/o DCD: We discard the whole DCDiff module and do not perform conditional diffusion denoising on auxiliary behavior representations based on the converted or unconverted behaviors signals.
- w/o TamGCN: We replace the target-guided attention mask GCN (TamGCN) with a standard LightGCN, which consequently disables the mechanism for suppressing interactions with low semantic relevance to the target behavior.
- w/o Sim: We ignore the semantic correlations by discarding the Sim-Guided score in the SHF module and only employ Ctx-Guided score for behavior fusion.

Table 2
Overall performance comparison of DCDRec and baseline models on two datasets in terms of HR@K (R@K) and NDCG@K (N@K). The best results are bolded, and the second-best results are underlined.

Method	Taobao				Tmall				Jdata			
	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20
MF-BPR	0.0076	0.0095	0.0036	0.0155	0.0230	0.0316	0.0124	0.0144	0.1850	0.2192	0.1238	0.1325
NeuMF	0.0236	0.0472	0.0128	0.0152	0.0124	0.0237	0.0062	0.0084	0.2090	0.2461	0.1410	0.1504
LightGCN	0.0411	0.0822	0.0240	0.0266	0.0393	0.0538	0.0209	0.0243	0.2252	0.2825	0.1436	0.1582
RGCN	0.0215	0.0430	0.0104	0.0125	0.0316	0.0489	0.0157	0.0198	0.2406	0.3418	0.1444	0.1588
GNMR	0.0368	0.0736	0.0216	0.0263	0.0393	0.0619	0.0193	0.0247	0.3068	0.3694	0.1581	0.1944
NMTR	0.0282	0.0564	0.0137	0.0303	0.0517	0.0847	0.0250	0.0330	0.3142	0.4086	0.1717	0.1966
MBGCN	0.0509	0.0752	0.0294	0.0350	0.0549	0.0799	0.0285	0.0345	0.2803	0.3603	0.1572	0.1790
CRGCN	0.0855	0.1211	0.0439	0.0676	0.0840	0.1238	0.0442	0.0540	0.5001	0.6190	0.2914	0.3225
MB-CGCN	0.1233	0.1640	0.0677	0.0860	0.0984	0.1396	0.0558	0.0859	0.4349	0.4423	0.2758	0.2894
MB-HGCN	0.1299	0.1868	0.0690	0.0885	0.1461	0.2072	0.0770	0.0920	0.5338	0.6450	0.3238	0.3533
Mule	0.1923	0.2475	0.1110	0.1249	0.2114	0.2685	0.1169	0.1308	0.5850	0.6716	0.4091	0.4283
COPF	0.1552	0.2058	0.0838	0.0901	0.1755	0.2235	0.0967	0.1060	0.4131	0.5391	0.2406	0.2724
CMC-GCN	0.1987	0.2564	0.1064	0.1231	0.1725	0.2351	0.0953	0.1102	0.5395	0.6362	0.3285	0.3530
DCDRec	**0.2091**	**0.2611**	**0.1200**	**0.1331**	**0.2229**	**0.2763**	**0.1267**	**0.1393**	**0.5925**	**0.6795**	**0.4181**	**0.4312**
Impr.%	5.23%	1.83%	9.19%	7.25%	5.44%	2.91%	8.38%	6.50%	1.28%	1.16%	2.20%	0.67%

Table 3
Performance comparison of different DCDRec variants on all datasets.

Method	Taobao		Tmall		Jdata	
	H@10	N@10	H@10	N@10	H@10	N@10
w/o.Cond	0.2042	0.1161	0.2176	0.1215	0.5892	0.4112
w/o.Dist	0.2051	0.1168	0.2161	0.1201	0.5840	0.4097
w/o.Diff	0.1985	0.1143	0.2131	0.1185	0.5812	0.4062
w/o.DCD	0.1943	0.1128	0.2117	0.1172	0.5801	0.4041
w/o.TamGCN	0.1911	0.1012	0.2002	0.1071	0.5689	0.3901
w/o.Sim	0.2045	0.1185	0.2146	0.1197	0.5885	0.4102
w/o.Freq	0.2078	0.1190	0.2207	0.1256	0.5911	0.4134
DCDRec	**0.2091**	**0.1200**	**0.2229**	**0.1267**	**0.5925**	**0.4181**

- w/o Freq: We ignore the frequency characteristics by discarding the frequency-domain branch in SHF module and only employ time-domain weighted fusion.

The results in Table 3 demonstrate that all components play essential roles in enhancing the effectiveness of our proposed model. To be specific, omitting the unconverted behavior as a conditional signal (w/o Cond) causes a significant performance drop across all datasets. This demonstrates that conditioning the reverse diffusion process on this signal is essential as it helps filter out irrelevant information from the auxiliary behavior, and preserves only the user intent that positively contributes to conversions towards the target behavior. Moreover, removing the distillation mechanism (w/o Dist) leads to a performance decline as well. This confirms its importance in aligning the diffused embeddings with the converted behavior embeddings, which effectively alleviates the semantic disparity between auxiliary and target behaviors during diffusion. Additionally, replacing the iterative diffusion denoising process (w/o Diff) performs worse than DCDRec, demonstrating that the iterative diffusion process provides superior denoising capabilities compared to single-step distillation, which justifies its necessity in modeling complex behavior distributions. It is worth noting that discarding the whole diffusion module (w/o DCD) results in a substantial performance decline, indicating that the diffusion mechanism is indispensable for refining auxiliary behavior embeddings and capturing true user preferences. When TamGCN is replaced with LightGCN (w/o.TamGCN), the model's performance drops notably, showing the importance of modulating attention to emphasize informative relations while suppressing redundant or weakly correlated ones. Neglecting the semantic similarity scores (w/o.Sim) causes a performance loss, verifying that the semantic similarity provides a critical perspective for adaptive multi-behavior fusion. Finally, excluding the frequency-domain branch (w/o.Freq) degrades performance, confirming that integrating complementary time- and frequency-domain information is essential for robust and accurate recommendations.

5.4. Sensitivity analysis of GCN layer

To investigate how the GCN layer number of LightGCN and TamGCN affect the model performance, we alter the layer number of LightGCN and TamGCN layers from {1, 2, 3} and {1, 2, 3, 4, 5}, respectively. The results are shown in Fig. 2, from which we can observe that the optimal value for $L_{LightGCN}$ is generally small (1 for Taobao and Tmall, 2 for Jdata), with performance tending to decline as depth increases. This is primarily because the over-smoothing effect in standard graph convolutions limits the discriminative power of representations. In contrast, performance consistently improves with the increase of L_{TamGCN}, peaking at 4 layers for Taobao/Tmall and 5 layers for Jdata. The contrast demonstrates that by filtering noise through the target-guided attention mask, TamGCN can capture high-order behavioral correlations more effectively as compared with standard GCNs, thereby significantly enhancing recommendation performance.

5.5. Parameter analysis (RQ4)

We further conduct experiments to investigate the impact of two hyperparameters, i.e., the diffusion loss weight λ and the quantile threshold τ of TamGCN, in our proposed approach.

Impact of the coefficient of diffusion loss. To evaluate the effectiveness of the distillation conditional diffusion module, we conduct experiments by varying the diffusion loss weight λ in $\{0.05, 0.1, 0.3, 0.5, 0.8\}$. As shown in Fig. 3, the performance of DCDRec on the Taobao dataset improves gradually with increasing λ from 0.05 to 0.5, and reaches a peak when λ equals to 0.5. After that, the performance begins to show a declining trend. The reason is that a too small λ might result in failing to effectively suppress noise in the auxiliary behavior and properly align with the target behavior representation. When λ begins too large, the diffusion loss would dominate the optimization objective and degrade the model performance.

It is worth noting that the impact of parameter λ over model performance on the Tmall dataset exhibits a similar trend as on the Taobao dataset. However, a key distinction lies in the optimal λ value as the performance on Tmall peaks at a much smaller λ than on Taobao. This is likely because auxiliary behaviors in the Tmall dataset contain less noise and are more strongly correlated with the target behaviors compared to those in Taobao.

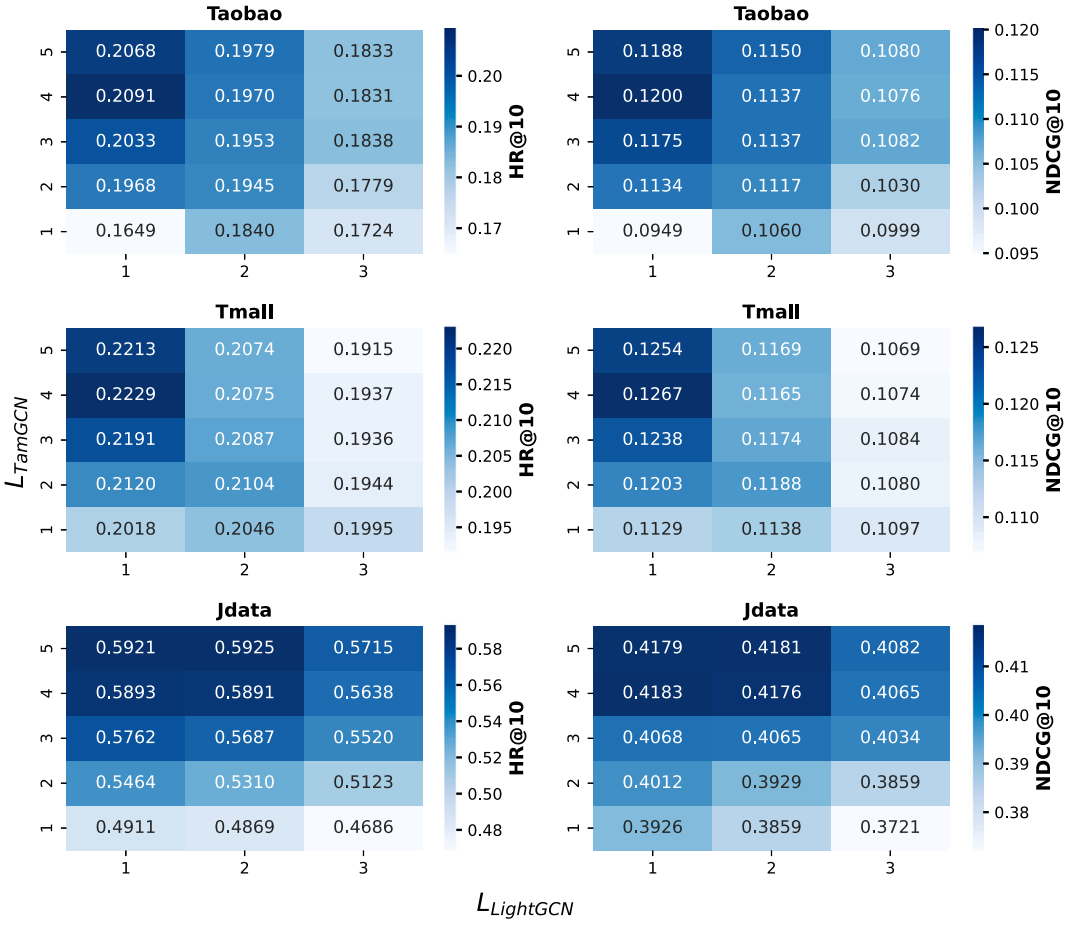

Fig. 2. Effect of $L_{LightGCN}$ and L_{TamGCN} of DCDRec, where $L_{LightGCN}$ and L_{TamGCN} are the numbers of LightGCN and TamGCN layers, respectively.

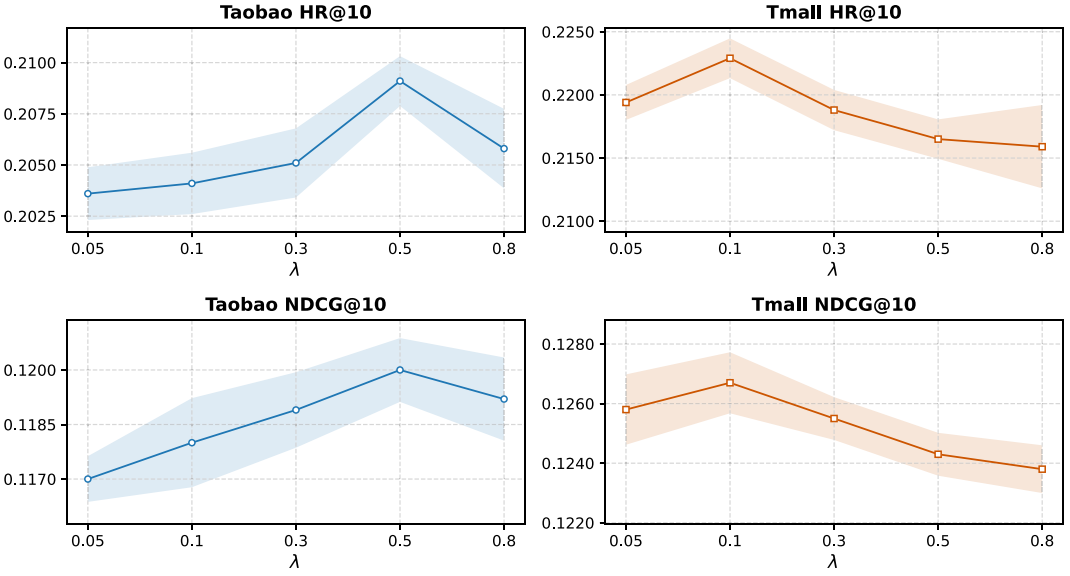

Fig. 3. Impact of the coefficient of the diffusion loss.

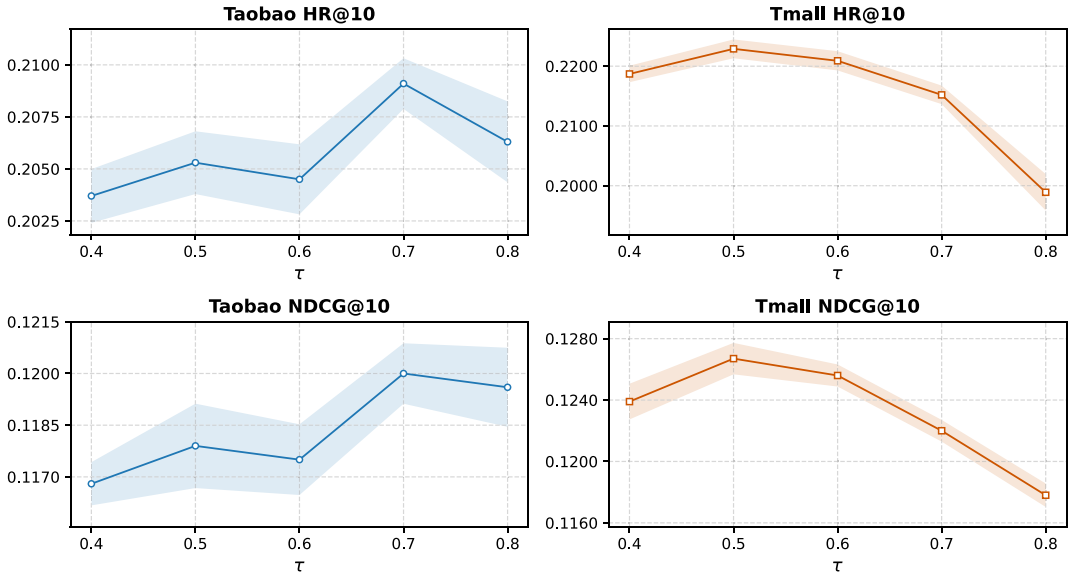

Fig. 4. Impact of the quantile threshold of TamGCN.

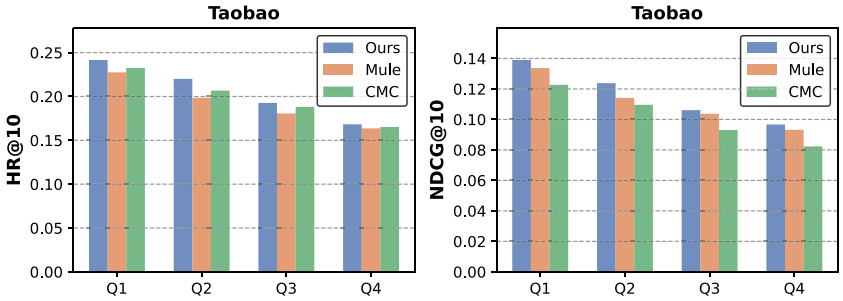

Fig. 5. Model performance with respect to different interaction density degrees.

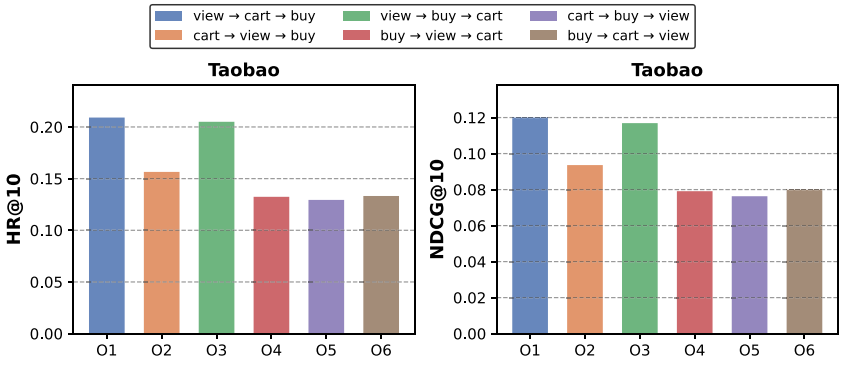

Fig. 6. Model performance with different behavior chain order.

Impact of the quantile threshold of TamGCN. To investigate the impact of the quantile threshold of TamGCN on model performance, we tune τ in {0.4, 0.5, 0.6, 0.7, 0.8 }. The results are shown in Fig. 4. We can observe that the performance first rises gradually and reaches a peak when τ equals to 0.7 on the Taobao dataset and 0.5 on the Tmall dataset. If we continue to raise the value of τ, the performance starts to drop considerably.

It is worth noting that the value of the threshold τ is closely related to the conversion rate (noise level) of the dataset. For example, for a high conversion rate (e.g., Tmall), we observe a relatively small τ value ($\tau = 0.5$). This is because a high conversion rate indicates that more interactions are reliable in the auxiliary behavior, thus necessitating a smaller value of τ to preserve more information from the auxiliary behavior. On the contrary, when the conversion rate is low (e.g., Taobao), there are more unreliable interactions, which necessitate a larger value of τ to suppress irrelevant signals in the auxiliary behavior.

5.6. Performance with different interaction density degrees

To evaluate the performance of our method across different user interaction densities, we conduct comparisons with two baselines (i.e., Mule and CMC-GCN) on the Taobao dataset. Specifically, we divide users in the test set into four groups based on the number of interactions: Q1 (highest 25% interactions), Q2 (25–50%), Q3 (50–75%), and Q4 (lowest 25%). The experimental results are presented in Fig. 5. We can observe that both our model DCDRec and all baselines demonstrate a general

Table 4
Model computational cost with running time (seconds).

Method	Taobao		Tmall		Jdata	
	Training	Inference	Training	Inference	Training	Inference
Mule	2.87	2.01	13.05	4.05	18.11	2.07
COPF	16.17	11.12	62.35	97.61	68.35	234.14
CMC-GCN	8.54	5.13	15.23	9.71	17.46	11.79
DCDRec	19.22	24.19	78.72	104.56	138.45	75.53

decline in performance as the interaction density decreases (from Q1 to Q4).

It is worth noting that DCDRec consistently outperforms both baselines across all user groups, demonstrating its superior robustness under varying interaction density degrees. This superiority may be attributed to the model's capability to mitigate semantic disparity by aligning auxiliary behaviors with target behaviors through a distillation conditional diffusion process. In addition, the hierarchical fusion mechanism further enhances performance by capturing semantic complementarities and spectral dynamics to adaptively integrate diverse behavioral signals.

5.7. Performance with different behavior chain order

To investigate the impact of behavior order on DCDRec, we evaluate six different permutations of the behavior chain on the Taobao dataset, as illustrated in Fig. 6. The results show that the default behavior chain [view → cart → buy] yields the best performance across all metrics. In contrast, placing the target behavior "buy" at the start of the chain (e.g., [buy → view → cart]) or reversing the order leads to a substantial performance drop. This demonstrates that the sequential dependency reflecting the natural progression of user decision-making is a critical factor for the effectiveness of DCDRec.

The performance disparity is primarily due to the cascading representation learning mechanism within DCDRec. Following the natural order [view → cart → buy] allows the model to build a broad interest foundation from high-frequency "view" data, providing a rich context for sparse, high-intent behaviors. Conversely, disrupting this order forces the model to initialize embeddings from sparse signals (like "buy"), failing to capture the natural evolution of user intent and hindering effective knowledge transfer across behavior-specific graphs.

5.8. Training time and inference latency analysis

To evaluate training time and inference latency, we compare the per-epoch time consumption of DCDRec with the state-of-the-art multi-behavior methods under same experimental conditions. The results are shown in Table 4. Specifically, Mule achieves the highest efficiency because it adopts a parallel multiplex graph structure, which allows for independent processing of different behaviors and avoids the recursive overhead of cascading or iterative denoising. CMC-GCN also shows competitive speed due to its streamlined cascading operations. In contrast, COPF incurs significant computational costs, especially during the inference stage on the large-scale Jdata dataset. This is primarily due to its complex behavior fitting experts in the prediction layer. While our DCDRec introduces additional training time primarily due to the iterative denoising steps in DCDiff, it demonstrates a significant efficiency advantage over COPF as the data scale increases. On the Jdata dataset, DCDRec's inference time is remarkably lower than that of COPF. This efficiency stems from the architectural separation: DCDRec integrates complex modeling into the embedding generation phase, while the final recommendation relies on efficient vector inner-product operations. This design allows DCDRec to maintain high scalability for online serving while capturing sophisticated behavioral patterns.

5.9. Qualitative analysis

To verify the quality of behavioral representations learned by our proposed DCDRec, we visualized user and item embeddings from the Taobao dataset by performing dimensionality reduction using t-SNE, as shown in Fig. 7. The embedding distributions compare the auxiliary behavior (View) and target behavior (Buy) before and after the diffusion process. We can observe that before the diffusion process, raw auxiliary embeddings exhibit a scattered distribution and significant overlap with target embeddings. This suggests that original interaction data suffers from substantial behavioral noise and semantic disparity, leading to representation degradation. After the diffusion process, the refined auxiliary embeddings are found to form distinct and compact semantic clusters. Instead of simply fitting the distribution of target behavior, these denoised representations demonstrate a structured alignment with conversion-oriented semantics. These results indicate that DCDRec effectively filters stochastic noise and distills informative behavioral priors, thereby bridging the semantic gap between heterogeneous behaviors.

5.10. Discussion

Beyond its technical performance, DCDRec is designed with an emphasis on data reliability and deployment feasibility. By leveraging the DCDiff and TamGCN modules to filter out stochastic noise in auxiliary behaviors, our model effectively mitigates bias propagation and aligns recommendations more closely with users' real purchase intent. Furthermore, the competitive latency, as shown in Table 4, underscores its feasibility for practical application.

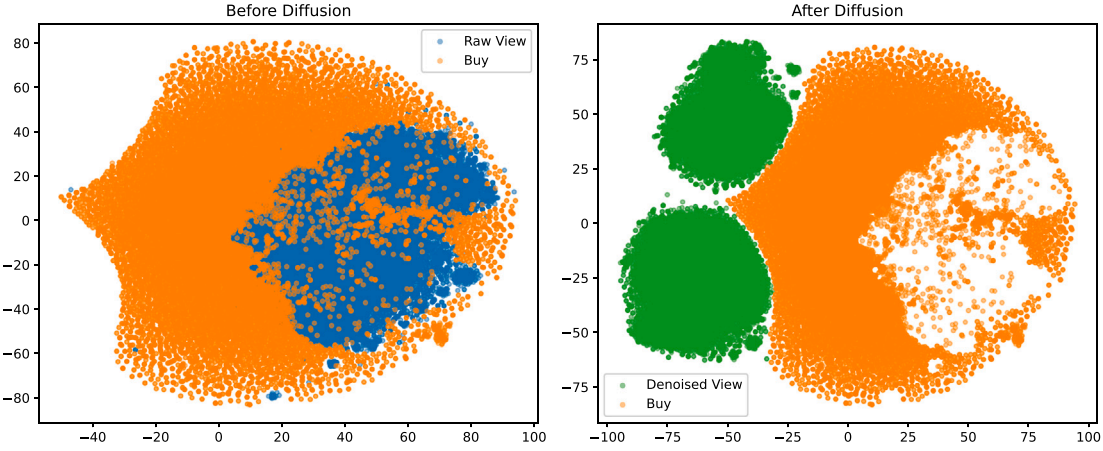

Fig. 7. Visualization of embedding distributions on Taobao dataset.

6. Conclusion

In this work, we propose a novel model named Distillation Conditional Diffusion with Spectral-enhanced Hierarchical Fusion (DCDRec) for multi-behavior recommendation. To tackle the semantic disparity among behaviors, DCDRec introduces a Robust Residual Cascading Graph Convolutional Network (RRC-GCN) to effectively model diverse and interrelated interactions with distinct conversion implications. Specifically, it leverages a target-guided attention mask mechanism to prune redundant structures and employs a distillation conditional diffusion process to align auxiliary behavioral patterns with target representations. Moreover, to handle the unstable fluctuations within behavioral signals, we develop a Spectral-enhanced Hierarchical Fusion (SHF) module to adaptively integrate multi-behavior information, which jointly captures hierarchical correlations, semantic complementarities, and spectral dynamics across behavioral embeddings. Extensive experiments on three real-world datasets demonstrate that DCDRec is consistently superior to all state-of-the-art baseline methods, achieving relative gains of up to 5.44% in HR@10 and 9.19% in NDCG@10. While DCDRec performs well in e-commerce contexts, its generalizability to other recommendation domains such as social networks or content platforms, where behavior hierarchies differ significantly, remains to be further explored. In the future, we will extend DCDRec under different recommendation domains and integrate it with large language models to improve its generalizability in modeling multi-behavior preferences.

CRediT authorship contribution statement

Xiaofei Zhu: Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization. **Peng Shan:** Writing – original draft, Visualization, Validation, Software, Methodology, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (62472059), the Science and Technology Innovation Key R&D Program of Chongqing (CSTB2024TIAD-STX0027), the Chongqing Talent Plan Project, China (CSTC2024YCJH-BGZXM0022), and the Open Research Fund of Key Laboratory of Cyberspace Big Data Intelligent Security, Ministry of Education (CBDIS202403).

Data availability

Data will be made available on request.

References

[1] Z. Cheng, J. Dong, F. Liu, L. Zhu, X. Yang, M. Wang, Disentangled cascaded graph convolution networks for multi-behavior recommendation, ACM Trans. Recomm. Syst. (2024) 31:1–31:27.
[2] Z. Cheng, S. Han, F. Liu, L. Zhu, Z. Gao, Y. Peng, Multi-behavior recommendation with cascading graph convolution networks, in: Proceedings of the ACM Web Conference 2023, 2023, pp. 1181–1189.
[3] Y. Dang, E. Yang, G. Guo, L. Jiang, X. Wang, X. Xu, Q. Sun, H. Liu, Uniform sequence better: time interval aware data augmentation for sequential recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2023, pp. 4225–4232.
[4] J. Ding, G. Yu, X. He, Y. Quan, Y. Li, T.-S. Chua, D. Jin, J. Yu, Improving implicit recommender systems with view data, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), 2018, pp. 3343–3349.
[5] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, D. Jin, Neural multi-task recommendation from multi-behavior data, in: Proceedings of the IEEE International Conference on Data Engineering, 2019, pp. 1554–1557.
[6] G. Guo, H. Qiu, Z. Tan, Y. Liu, J. Ma, X. Wang, Resolving data sparsity by multi-type auxiliary implicit feedback for recommender systems, Knowl.-Based Syst. (2017) 202–207.
[7] L. Guo, L. Hua, R. Jia, B. Zhao, X. Wang, B. Cui, Buying or browsing?: predicting real-time purchasing intent using attention-based deep network with multiple behavior, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019, pp. 1984–1992.
[8] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, M. Wang, Lightgcn: simplifying and powering graph convolution network for recommendation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 639–648.
[9] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T.-S. Chua, Neural collaborative filtering, in: Proceedings of the 26th International World Wide Web Conference, 2017, pp. 173–182.
[10] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, in: Advances in Neural Information Processing Systems, 2020, pp. 6840–6851.
[11] Y. Jiang, Y. Yang, L. Xia, C. Huang, Diffkg: knowledge graph diffusion model for recommendation, in: Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM), 2024, pp. 313–321.
[12] B. Jin, C. Gao, X. He, D. Jin, Y. Li, Multi-behavior recommendation with graph convolutional networks, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 659–668.
[13] D.S. Khafaga, A. Ibrahim, E.-S.M. El-Kenawy, A.A. Abdelhamid, F.K. Karim, S. Mirjalili, N. Khodadadi, W.H. Lim, M.M. Eid, M.E. Ghoneim, An Al-Biruni Earth radius optimization-based deep convolutional neural network for classifying monkeypox disease, Diagnostics (2022) 2892.
[14] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, in: 3rd International Conference on Learning Representations (ICLR), 2015.
[15] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: 5th International Conference on Learning Representations (ICLR), 2017.
[16] A. Krohn-Grimberghe, L. Drumond, C. Freudenthaler, L. Schmidt-Thieme, Multi-relational matrix factorization using Bayesian personalized ranking for social network data, in: Proceedings of the 5th ACM International Conference on Web Search and Data Mining (WSDM), 2012, pp. 173–182.
[17] S. Lee, G. Ko, H.-J. Song, J. Jung, Mule: multi-grained graph learning for multi-behavior recommendation, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM), 2024, pp. 1163–1173.
[18] X. Li, C. Fu, Z. Zhao, G. Zheng, C. Huang, J. Dong, Y. Yu, Dual-channel multiplex graph neural networks for recommendation, arXiv preprint arXiv:2403.11624, 2024.
[19] Z. Li, L. Xia, H. Hua, S. Zhang, S. Wang, C. Huang, Diffgraph: heterogeneous graph diffusion model, in: Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining (WSDM), 2025, pp. 40–49.
[20] Z. Li, L. Xia, C. Huang, Recdiff: diffusion model for social recommendation, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM), 2024, pp. 1346–1355.
[21] R.K. Ong, A.W.H. Khong, Spectrum-based modality representation fusion graph convolutional network for multimodal recommendation, in: Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining, 2025, pp. 773–781.
[22] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, Bpr: Bayesian personalized ranking from implicit feedback, in: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, 2009, pp. 452–461.
[23] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: Proceedings of the European Semantic Web Conference, 2018, pp. 593–607.
[24] A.P. Singh, G.J. Gordon, Relational learning via collective matrix factorization, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008, pp. 650–658.
[25] L. Tang, B. Long, B.-C. Chen, D. Agarwal, An empirical study on recommendation with multiple types of feedback, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 283–292.
[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30 (NeurIPS), 2017, pp. 5998–6008.
[27] W. Wang, Y. Xu, F. Feng, X. Lin, X. He, T.-S. Chua, Diffusion recommender model, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023, pp. 832–841.
[28] L. Xia, C. Huang, Y. Xu, P. Dai, M. Lu, L. Bo, Multi-behavior enhanced recommendation with cross-interaction collaborative relation modeling, in: Proceedings of the IEEE International Conference on Data Engineering, 2021, pp. 1931–1936.
[29] L. Xia, C. Huang, Y. Xu, P. Dai, B. Zhang, L. Bo, Multiplex behavioral relation learning for recommendation via memory augmented transformer network, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 2397–2406.
[30] L. Xia, Y. Xu, C. Huang, P. Dai, L. Bo, Graph meta network for multi-behavior recommendation, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 757–766.
[31] M. Yan, Z. Cheng, C. Gao, J. Sun, F. Liu, F. Sun, H. Li, Cascading residual graph convolutional network for multi-behavior recommendation, ACM Trans. Inf. Syst. (2023) 10:1–10:26.
[32] M. Yan, Z. Cheng, J. Sun, F. Sun, Y. Peng, Mb-hgcn: A hierarchical graph convolutional network for multi-behavior recommendation, arXiv preprint arXiv:2306.10679, 2023.

[33] Y. Yin, X. Zhu, K. Huang, W. Wang, Y. Zhang, P. Wang, Y. Fan, J. Guo, Cmc-gcn: consistent multi-granularity cascading graph convolution network for multi-behavior recommendation, Neurocomputing (2025) 130952.
[34] C. Zhai, C. Meng, Y. Yang, K. Zhang, X. Zhao, X. Li, Combinatorial optimization perspective based framework for multi-behavior recommendation, in: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2025, pp. 1891–1902.
[35] W. Zhang, J. Mao, Y. Cao, C. Xu, Multiplex graph neural networks for multi-behavior recommendation, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM), 2020, pp. 2313–2316.

Author biography

Prof. Dr. Xiaofei Zhu is a full professor at College of Computer Science and Engineering, Chongqing University of Technology. He received his PhD degree at the Institute of Computing Technology, Chinese Academy of Science (ICT-CAS) in 2012. Then he spent four years as a Postdoctoral Research Fellow at the L3S Research Center, Leibniz University Hannover. His research interests include web search, data mining and machine learning, and he has published more than 30 papers in international conferences and journals, including the top conferences like SIGIR, WWW, CIKM, TKDE, etc. He has won the Best Paper Awards of CIKM (2011). He serves as area chair, program committees and editorial board of numerous international conferences and journals, including SIGIR, AAAI, CIKM, etc.

Peng Shan is a Master Candidate at College of Computer Science and Engineering, Chongqing University of Technology. He received the B.S. degree in Computer Science and Technology from Chengdu Jincheng College in 2022. His main research interest focuses on machine learning, text mining and recommendation.