

Unleashing the Potential of Neighbors: Diffusion-based Latent Neighbor Generation for Session-based Recommendation

Yuhan Yang

University of Electronic Science and
Technology of China
Chengdu, China
y.yuhan2000@gmail.com

Jie Zou*

University of Electronic Science and
Technology of China
Chengdu, China
jie.zou@uestc.edu.cn

Guojia An

University of Electronic Science and
Technology of China
Chengdu, China
an_guojia@163.com

Jiwei Wei

University of Electronic Science and
Technology of China
Chengdu, China
mathematic6@gmail.com

Yang Yang

University of Electronic Science and
Technology of China
Chengdu, China
yang.yang@uestc.edu.cn

Heng Tao Shen

University of Electronic Science and
Technology of China
Chengdu, China
shenhengtao@hotmail.com

Abstract

Session-based recommendation aims to predict the next item that anonymous users may be interested in, based on their current session interactions. Recent studies have demonstrated that retrieving neighbor sessions to augment the current session can effectively alleviate the data sparsity issue and improve recommendation performance. However, existing methods typically rely on explicitly observed session data, neglecting latent neighbors - not directly observed but potentially relevant within the interest space - thereby failing to fully exploit the potential of neighbor sessions in recommendation.

To address the above limitation, we propose a novel model of diffusion-based latent neighbor generation for session-based recommendation, named **DiffSBR**. Specifically, DiffSBR leverages two diffusion modules, including retrieval-augmented diffusion and self-augmented diffusion, to generate high-quality latent neighbors. In the retrieval-augmented diffusion module, we leverage retrieved neighbors as guiding signals to constrain and reconstruct the distribution of latent neighbors. Meanwhile, we adopt a training strategy that enables the retriever to learn from the feedback provided by the generator. In the self-augmented diffusion module, we explicitly guide the generation of latent neighbors by injecting the current session's multi-modal signals through contrastive learning. After obtaining the generated latent neighbors, we utilize them to enhance session representations for improving session-based recommendation. Extensive experiments on four public datasets show that DiffSBR generates effective latent neighbors and improves recommendation performance against state-of-the-art baselines.

CCS Concepts

• Information systems → Recommender systems.

*Corresponding author.



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

KDD '26, Jeju Island, Republic of Korea

© 2026 Copyright held by the owner/author(s).

ACM ISBN 978-x-xxxx-xxxx-x/YYYY/MM

<https://doi.org/10.1145/nnnnnnnnnnnnnn>

Keywords

Session-based Recommendation, Diffusion Model, Multi-modal

ACM Reference Format:

Yuhan Yang, Jie Zou, Guojia An, Jiwei Wei, Yang Yang, and Heng Tao Shen. 2026. Unleashing the Potential of Neighbors: Diffusion-based Latent Neighbor Generation for Session-based Recommendation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/nnnnnnnnnnnnnn>

1 Introduction

Given the exponential growth of multimedia content, users face increasing challenges in finding information that aligns with their preferences [35, 55]. Fortunately, the emergence of recommender systems [2, 3, 38, 39, 56] has alleviated this issue. However, due to growing concerns over user privacy, it is often infeasible to access users' personal information and historical interactions. To address this issue, session-based recommendation (SBR) [4, 22, 43] has emerged as a promising solution. SBR aims to recommend the next item based solely on the interaction of anonymous users within a session.

In SBR, the lack of user profiles, coupled with short session lengths, intensifies data sparsity, thereby degrading overall recommendation performance. To alleviate this issue, recent studies primarily adopt retrieval-based neighbor methods [29, 51] to mine neighbor information and enhance the target session representation. These methods can be broadly categorized into similarity-based and co-occurrence-based approaches. The former (e.g., ICM-SR [25], DIDN [49], TASI-GNN [27], ECCL [1]) select semantically similar sessions via static similarity learning, while the latter (e.g., FGNN [30], MSGAT [29], DGNN [19], DIMO [51]) leverage item co-occurrence patterns to capture structural relationships.

Although these existing neighbor retrieval methods have made notable progress in the field of SBR, they still face inherent limitations. These methods are constrained by the scope of the dataset and only retrieve existing **observable neighbors**, which refer to *interest-aligned sessions explicitly contained in the dataset*. However, this paradigm overlooks **latent neighbors**, which are not recorded in the dataset and therefore cannot be discovered through retrieval methods but are still aligned with the user's underlying interests,

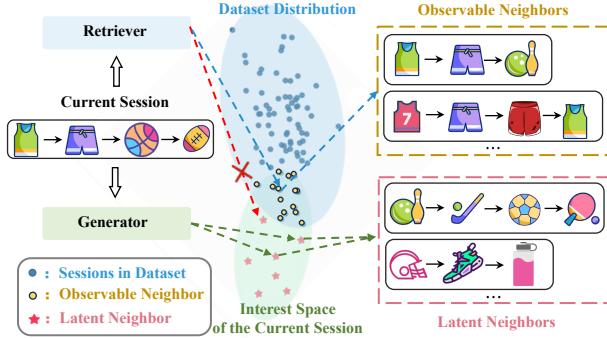


Figure 1: Retrieval-based method vs. generative method. The blue region represents the distribution of the dataset, while the green region denotes the full interest space of the current session. Retrieval-based method can only access interest-aligned neighbors within the intersection of these regions, thus being constrained by observed data. In contrast, the generative method overcomes this limitation by exploring the entire interest space, enabling the generation of potentially relevant but unobserved latent neighbors.

thereby limiting the expressiveness of neighbors and ultimately constraining recommendations. Specifically, as shown in Figure 1, we first project the user’s current session representation into a unified interest space. Under the retrieval paradigm, observable neighbors are obtained from samples that exist in the dataset (blue region in Figure 1). While such retrieved neighbors may offer partial coverage of the user’s interest space, they are fundamentally constrained by the boundaries of the dataset, thereby failing to access the full extent of the user’s interest space. In contrast, the generative paradigm models the potential distribution of user interests and expands the scope of neighbors (within the green region but outside the green-blue overlap in Figure 1). This approach enables active exploration of interest regions beyond the observed data, allowing the generation of latent neighbors that are inaccessible through traditional retrieval methods.

To overcome the inherent limitations of the aforementioned retrieval paradigm, we take an initial step toward transitioning from a retrieval-based framework to a generative paradigm, aiming to generate latent neighbors beyond the scope of observable neighbors. Inspired by the remarkable advances of diffusion models in modeling complex data distributions [6, 45], we adopt diffusion as the main generative mechanism in our framework. Despite their promising generative capabilities, diffusion models still face two key challenges in ensuring the quality and effectiveness of the generated latent neighbors: (1) How to effectively design guidance mechanisms to improve the quality of latent neighbors generated by diffusion models. (2) How to effectively integrate multi-modal information to enhance latent neighbors generation in the diffusion generation process.

To address the aforementioned two challenges, we propose a novel model of **Diffusion-based latent neighbor generation for Session-based Recommendation**, called **DiffSBR**, which guides the diffusion model to generate high-quality latent neighbors. Specifically, to tackle the first challenge, we proposed a novel module called **Retrieval-augmented Diffusion Module**, which controls

generation by incorporating the retrieved prior neighbor information during the diffusion generation process, so that latent neighbors can be generated in a targeted manner. Moreover, a training strategy is introduced in this module to reversely optimize the neighbor retriever using the loss signal from the neighbor generator, establishing a closed-loop collaboration between retrieval and generation. To address the second challenge above, we design a **Self-augmented Diffusion Module**, which integrates multi-modal information into the process of latent neighbor generation. Specifically, we use multi-modal information to guide generation and then conduct contrastive learning against the retrieval-augmented diffusion outputs to better integrate multi-modal semantics.

In this paper, our main contributions are summarized as follows:

- We perform a pioneering attempt to generate latent neighbors and highlight their importance, aiming to mitigate the inherent limitations of traditional retrieval-based methods in SBR.
- We propose a novel DiffSBR model, an effective framework that generates effective latent neighbors through a retrieval-augmented diffusion module and a self-augmented diffusion module, thereby improving recommendation performance.
- We conduct extensive experiments on four public datasets, demonstrating that our proposed DiffSBR model, not only significantly outperforms existing SBR methods, but also proves the effectiveness and necessity of generating latent neighbors.

2 Related Work

2.1 Session-based Recommendation

SBR models user preferences from short-term anonymous interactions. Early methods like FPMC [46] combine Markov chains with matrix factorization but fail to capture high-order dependencies. GRU4Rec [11] and NARM [17] improve sequential modeling via RNNs and attention. Recent GNN-based models (e.g., SR-GNN [40], TAGNN [47]) treat sessions as graphs to better capture complex item relations. However, they rely solely on intra-session information, ignoring valuable signals from neighboring sessions. To mitigate this, some methods retrieve neighbor sessions to enrich current session representations.

Retrieval-based neighbor methods in SBR. These methods aim to integrate cross-session neighbor information to enhance session representations under sparse interactions. Existing studies can be broadly categorized into two main types. The first focuses on similarity-based retrieval methods [1, 29], which identify relevant neighbor sessions by computing the similarity. For example, ICM-SR [25] employs an intention-guided neighbor detector to locate relevant sessions, while DIDN [49] utilizes a dynamic intention-aware module to retrieve semantically similar sessions. The second line of work leverages co-occurrence relationships [37, 51], constructing global graphs based on item co-occurrence to capture pairwise transitions across sessions. For instance, CGL [54] builds a global graph to model inter-session correlations to enhance item representations. MSGAT [29] further constructs a session-level relation graph and incorporates an intent-aware collaboration module to refine the session representation.

Although the aforementioned retrieval-based methods can achieve effectiveness, they only retrieve observable neighbors from the dataset, which consequently limits the quality of the neighbors. In contrast, we exploit the generative capability of diffusion models to generate latent neighbors, thereby uncovering semantically relevant neighbors that are not explicitly present in the data.

Multi-modal-based methods in SBR. Since user interests are often driven by multi-modal content, relying solely on ID features is insufficient to reflect true preferences. Recent works leverage rich item features to improve user intent modeling. For example, CoHHN [52] incorporates price as a key modality; MMSBR [50] combines item text and images; LLM4SBR [28] utilizes textual descriptions and prompts large language models for intent inference; and DIMO [51] decouples and fuses ID and multi-modal features. Distinct from these approaches, our method is the first to explicitly integrate multi-modal signals into the neighbor generation process in SBR, to the best of our knowledge.

2.2 Diffusion Models in Recommendation

Diffusion models have recently emerged as a promising generative framework for recommendation, offering strong capacity for uncertainty modeling and flexible preference generation. Early work like DiffRec [34] applies diffusion to user preference modeling and item generation. DiffuRec [18] and DiQDiff [24] extend this idea to sequential recommendation via reverse diffusion. Further, DCASR [10] and DiffuASR [20] explore diffusion-based augmentation for improving sequence representation. Beyond sequences, DiffKG [15] and DiffMM [14] introduce diffusion into structured data, including knowledge graphs and multimodal interaction graphs. Recently, DDRM [53] and MCDRec [23], employ conditional diffusion to integrate user preferences into the generation process, enhancing personalization and semantic alignment. Although prior efforts have validated the effectiveness of diffusion models, most methods focus on target item generation or sequence augmentation. Instead, in this paper, we take the first step to generate latent neighbor information in SBR, which remains largely underexplored by previous work.

3 Problem Formulation

Let $\mathcal{V} = \{v_1, \dots, v_i, \dots, v_n\}$ denote the set of all items in the dataset, where n is the total number of unique items. Each item $v_i \in \mathcal{V}$ consists of an identifier v_i^{id} and multi-modal content features v_i^{mo} , i.e., $v_i = \{v_i^{id}, v_i^{mo}\}$. In this work, the multi-modal content features v_i^{mo} include textual and visual modalities: $v_i^{mo} = \{v_i^{txt}, v_i^{img}\}$, where $v_i^{txt} = \{w_1, w_2, \dots, w_q\}$ denotes a sequence of q words describing the item's title and brand name, and v_i^{img} represents the corresponding image of item v_i . Let $\mathcal{S} = \{s_1, \dots, s_j, \dots, s_{|\mathcal{S}|}\}$ denote the set of all sessions, where $|\mathcal{S}|$ is the total number of sessions. Each session s_j is an ordered sequence from an anonymous user within a short time period, formally defined as: $S_j = [v_1, v_2, \dots, v_m]$, where m is the length of the current session. Given the current session history $[v_1, v_2, \dots, v_m]$, the objective of SBR is to predict the next item v_{m+1} that the user is most likely to interact with.

4 Methodology

In this section, we provide a detailed description of our proposed DiffSBR framework, as illustrated in Figure 2. Each input session is first encoded via an ID and a multi-modal session representation module, resulting in a session ID embedding and a session modality embedding. These representations are then fed into the retrieval-augmented diffusion module and the self-augmented diffusion module to generate latent neighbors, thereby enhancing the current session for the final recommendation task.

4.1 ID and Multi-modal Session Representation

Here, we aim to obtain the session ID embedding and session modality embedding for each session separately through three main steps.

4.1.1 Initialization of Item Embeddings. Given the differences in the presentation of various modalities, we adopt specific encoding methods to convert raw modality data into vector representations. For each item v_i , we construct three types of embeddings: a structured ID embedding, a semantic embedding from textual descriptions, and a visual embedding derived from item images.

For structured ID embedding v_i^{id} of each item v_i , we follow common practices in prior work (e.g., [17, 40]) to construct an ID embedding table $\mathbf{E}^{id} \in \mathbb{R}^{n \times d}$, where each row corresponds to a randomly initialized ID embedding $e_i^{id} \in \mathbb{R}^d$ of a specific item.

There exists a substantial semantic gap between textual and visual modalities. Similar to Zhang et al. [51], we address this issue by transforming the visual modality into a textual form, thereby aligning heterogeneous modalities into a shared semantic space. Specifically, we employ GoogLeNet [31] to predict the top-2 category labels for each image of items, which are concatenated to form a pseudo-textual representation $v_i^{txt} = \{w'_1, w'_2, \dots, w'_o\}$. This is further concatenated with the original item description $v_i^{txt} = \{w_1, w_2, \dots, w_q\}$ to construct a unified multi-modal input $v_i^{mo} = \{w_1, \dots, w_q, w'_1, \dots, w'_o\}$. We feed v_i^{mo} into a pre-trained BERT [5] model to obtain contextualized token embeddings $\{e_1, e_2, \dots, e_{l+o}\}$, and apply average pooling to initialize the multi-modal item representation $e_i^{mo} \in \mathbb{R}^d$, computed as:

$$e_i^{mo} = \frac{1}{q+o} \sum_{r=1}^{q+o} e_r. \quad (1)$$

4.1.2 Graph-enhanced Item Embeddings. After obtaining the initialized item embeddings, we incorporate graph structure to further model transition dependencies and collaborative relationships. Following prior work [42, 51], we construct a directed item graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ based on co-occurrence patterns, where each node represents an item and the edge weight between two items reflects their co-occurrence frequency. From this graph, we obtain the corresponding adjacency matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, where $\mathbf{A}(i, j)$ denotes the edge weight from item v_i to item v_j . We also construct the degree matrix $\mathbf{D} \in \mathbb{R}^{n \times n}$, where $\mathbf{D}(i, i)$ is the out-degree of item v_i , defined as the sum of its outgoing edge weights.

We then apply an L -layer Graph Convolutional Network (GCN) to perform message aggregation. For both ID and multi-modal embeddings (denoted as e_i^{id} and e_i^{mo}), the l -th layer update is defined as:

$$x_{v_i^c}^{(l)} = \text{Norm} \left(\mathbf{D}^{-1} \mathbf{A} x_{v_i^c}^{(l-1)} \mathbf{W}^{(l-1)} \right), \quad (2)$$

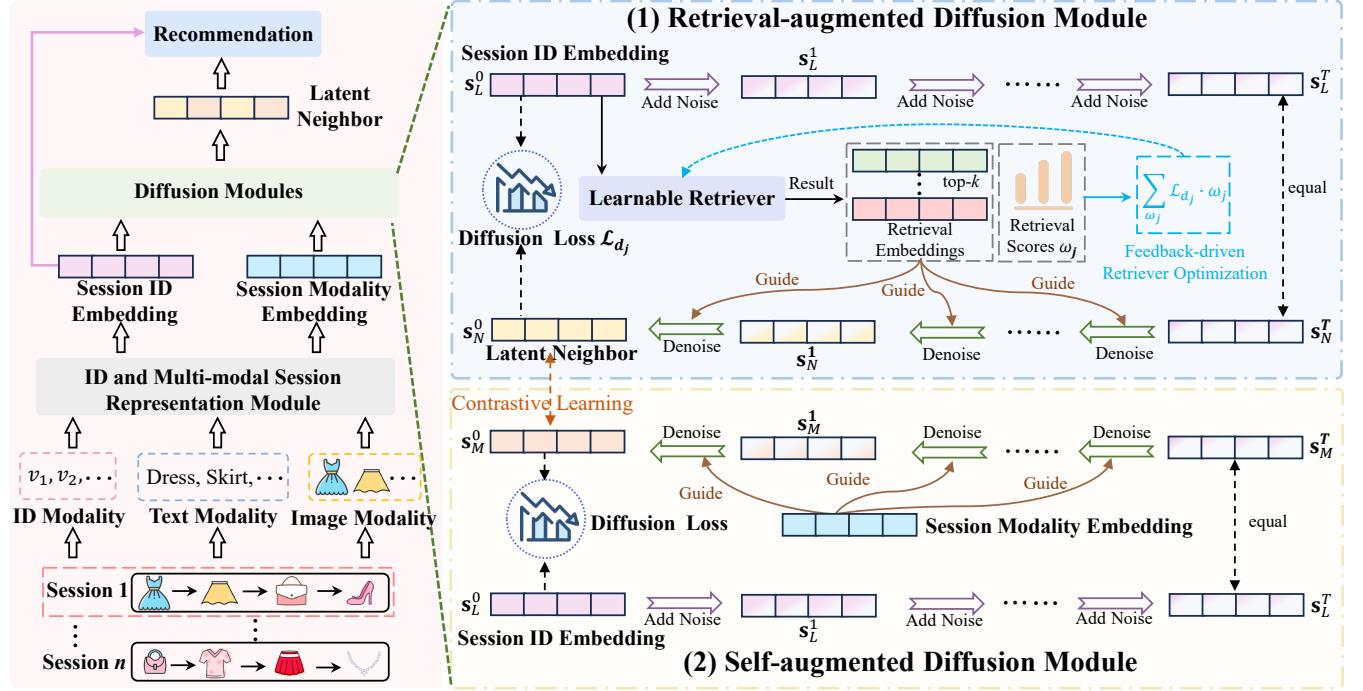


Figure 2: The overall framework of DiffSBR. The left part is the overall framework, and the right part represents the two main components of our diffusion module: (1) the retrieval-augmented diffusion module, which generates latent neighbors guided by retrieved priors; (2) the self-augmented diffusion module, which leverages the session’s own multi-modal information to improve the quality of latent neighbors.

where $c \in \{id, mo\}$, $\mathbf{x}_{v_i^c}^{(l)}$ denotes the item embedding at the l -th GCN layer, and $\mathbf{W}^{(l-1)} \in \mathbb{R}^{d \times d}$ is a learnable transformation matrix. $\text{Norm}(\cdot)$ represents a normalization function. The input to the GCN is the initialized item embedding: $\mathbf{x}_{v_i^id}^0 = \mathbf{e}_i^{id}$, $\mathbf{x}_{v_i^mo}^0 = \mathbf{e}_i^{mo}$. Finally, we aggregate the representations across all GCN layers to obtain the final graph-enhanced item embedding $\mathbf{x}_{v_i^c}$:

$$\mathbf{x}_{v_i^c} = \frac{1}{L+1} \sum_{l=0}^L \mathbf{x}_{v_i^c}^{(l)}. \quad (3)$$

4.1.3 Session Representation. To further obtain session-level representation, we first employ a contrastive alignment to encourage consistency between id and mo embeddings, which stabilizes multi-modal training, then we compute relevance scores over all items in the sequence, referring to Wu et al. [40]. To account for the varying importance of item embeddings, we introduce an attention-based aggregation [7] that adaptively weighs each item. The attention weights α_i are computed as:

$$\alpha_i = \sigma \left(\mathbf{W}_1 \mathbf{x}_{v_m^c} + \mathbf{W}_2 \mathbf{x}_{v_i^c} \right), \quad (4)$$

where \mathbf{W}_1 and \mathbf{W}_2 are learnable parameters, $\mathbf{x}_{v_m^c}$ is the last-clicked item in the sequence, and $\sigma(\cdot)$ denotes the sigmoid activation function. Then, the final session representation \mathbf{s}_c is obtained by aggregating item embeddings with the learned attention weights:

$$\mathbf{s}_c = \sum_{i=1}^m \alpha_i \cdot \mathbf{x}_{v_i^c}, \quad (5)$$

where $\mathbf{s}_c \in \{\mathbf{s}_{id}, \mathbf{s}_{mo}\}$.

4.2 Retrieval-augmented Diffusion Module

In this work, we adopt diffusion models as our generative backbone due to their superior training stability and high-quality sample generation compared to Generative Adversarial Networks (GANs) [8] and Variational Autoencoders (VAEs) [26], as demonstrated in prior studies [18, 24, 45]. Moreover, diffusion models have shown promising performance in recommendation tasks, making them a suitable choice for our generative framework.

To ensure that the diffusion model explores the interest space in a goal-directed manner, and to prevent it from drifting into semantically irrelevant or user-unrelated regions, we design a retrieval-augmented diffusion module. This module leverages retrieved prior knowledge to guide the diffusion process, enabling the generation of results that are not only semantically coherent but also highly aligned with the user’s current preferences.

4.2.1 Retrieval-based Prior Construction. To construct a neighbor set that provides prior guidance for the diffusion process, we first introduce a learnable retriever to select the top- k most relevant neighbors to the current session from the historical session database. Specifically, given a session \mathbf{s}_{id} , we compute its similarity score using the following equation:

$$\text{sim}^D = \mathcal{F}_{\text{score}} \left([\mathbf{s}_{id} \parallel \mathbf{s}^D] \right), \quad (6)$$

where $\mathcal{F}_{\text{score}}$ denotes a multi-layer perceptron, and $[\cdot \parallel \cdot]$ represents vector concatenation. Then we select the top- k candidate sessions

N_k with the highest similarity scores. Next, we apply the softmax function to normalize the similarity scores, in order to obtain the attention weight for each neighbor:

$$\omega_j = \frac{\exp(\text{sim}_j^D)}{\sum_{r=1}^k \exp(\text{sim}_r^D)}, \quad (7)$$

where $j \in \{1, \dots, k\}$. The neighbor session N_k , together with their corresponding score weights, is used as conditional information to guide the diffusion model in the downstream process.

4.2.2 Diffusion-based Neighbor Generation. After obtaining prior information to guide the diffusion process, we adopt a conditional DDPM [44] to train a denoising network, which progressively removes noise during inference to generate conditionally guided latent neighbors. Specifically, during training, a forward Markov chain is constructed to gradually add Gaussian noise to the original data sample, denoted as the clean session embedding $s_{id} = s_L^0$, over T time steps $t \in \{1, \dots, T\}$. This process eventually produces a noisy vector s_L^T that approximates a standard normal distribution. Each forward transition is defined as:

$$q(s_L^t | s_L^{t-1}) = \mathcal{N}(s_L^t; \sqrt{1 - \beta_t} s_L^{t-1}, \beta_t \mathbf{I}), \quad (8)$$

where $\beta_t \in (0, 1)$ is the noise schedule at time step t , and \mathbf{I} is the identity matrix. Let $\alpha_t = 1 - \beta_t$ and define the cumulative product as: $\bar{\alpha}_t = \prod_{i=1}^t \alpha_t$. Then, the noisy sample at step t can be directly derived from the clean embedding as:

$$s_L^t = \sqrt{\bar{\alpha}_t} s_L^0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, \quad (9)$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ [12, 36]. Following previous studies [24, 45, 53], we do not explicitly predict the added noise during training. Instead, we directly generate the latent neighbor representation s_N^0 at each timestep under the guidance of the semantic prior N_k , enabling the model to generate potential neighbors in a targeted manner:

$$s_N^0 = f_\theta(s_L^t, N_k, t), \quad (10)$$

where $f_\theta(\cdot)$ adopts an MLP architecture as in Mao et al. [24] and Wu et al. [41]. In our model, the reverse generation process is conceptualized as a conditional Gaussian distribution at each step:

$$p_\theta(s_N^{t-1} | s_N^t, N_k) = \mathcal{N}(s_N^{t-1}; \mu_\theta(s_N^t, N_k, t), \Sigma_\theta(s_N^t, N_k, t)), \quad (11)$$

where $\mu_\theta(\cdot)$ and $\Sigma_\theta(\cdot)$ are the learnable mean and covariance predicted by the network. Accordingly, the diffusion loss \mathcal{L}_d is defined as:

$$\mathcal{L}_d = \mathbb{E}_{t, s_L^0, \epsilon} \left[\|s_{id} - f_\theta(\sqrt{\bar{\alpha}_t} s_L^0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, N_k, t)\|^2 \right] \quad (12)$$

During inference, we employ a deterministic strategy to generate latent neighbors s_N^0 . Specifically, we first apply T' steps of forward corruption to the input session embedding s_{id} to obtain a noisy initialization. Then, starting from this point, we perform T' steps of reverse denoising conditioned on the retrieved neighbor prior N_k . In the deterministic reverse process, variance is omitted and the predicted mean is used directly:

$$s_N^{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} f_\theta(s_N^t, N_k, t) + \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} s_N^t, \quad (13)$$

where the model $f_\theta(\cdot)$ is encouraged to learn session-specific denoising strategies for each input. The final output s_N^0 is then used as the generated latent neighbor representation.

4.2.3 Feedback-driven Retriever Optimization. Our method is designed to enable the retriever to continually learn from the feedback of the generator, so as to retrieve neighbor sessions that are more beneficial for the neighbor generation task. To achieve this, we evaluate the effectiveness of each retrieved neighbor in guiding the diffusion process. Specifically, given the target session's ID-based representation s_{id} and a set of retrieved neighbors N_k , we compute the generator's loss \mathcal{L}_d through the diffusion process defined in section 4.2.2. A smaller value of \mathcal{L}_d indicates that the corresponding neighbor contributes more effectively to the generation of latent neighbors. Inspired by Lu and Liu [21], we introduce a relative ranking-based supervision mechanism: for two retrieved neighbors $s_i^D, s_j^D \in N_k$, if the diffusion loss under s_i^D as the condition is lower than that under s_j^D (i.e., $\mathcal{L}_{d_i} < \mathcal{L}_{d_j}$), then s_i^D is considered more useful for generation and should be assigned a higher retrieval score. To optimize the retriever according to the generator's preferences, we propose a training strategy that incorporates this supervision signal. Considering the large size of the candidate pool in practice, we improve efficiency by applying supervision only to the top- k retrieved neighbors. The corresponding loss function \mathcal{L}_r is defined as:

$$\mathcal{L}_r = \sum_{\omega_j \in \text{top-}k} \mathcal{L}_{d_j} \cdot \omega_j, \quad (14)$$

where ω_j denotes the softmax-normalized relevance score of neighbor s_j^D computed by the retriever, and \mathcal{L}_{d_j} is the diffusion loss incurred when using s_j^D as the generation condition.

4.3 Self-augmented Diffusion Module

To effectively incorporate modality-specific information into the diffusion process without interfering with the retrieval-augmented diffusion, we design a self-augmented diffusion module. This module conducts contrastive learning between the retrieval-augmented and self-augmented diffusion paths, thereby enhancing semantic alignment across modalities.

Specifically, we apply the forward diffusion process to the ID representation s_{id} by adding Gaussian noise, yielding noisy representations s_L^t at timestep t . During the reverse diffusion process, unlike the retrieval-augmented diffusion module, which performs denoising under the guidance of retrieved neighbors, the self-augmented diffusion module leverages the multi-modal representations of the current input itself to guide the denoising. The denoising is carried out by the network $f_\psi(\cdot)$, yielding the denoised embedding $s_M^t = f_\psi(s_L^t, s_{mo}, t)$. Similar to the retrieval-augmented diffusion module, we compute the diffusion loss \mathcal{L}_s for the self-augmented module as follows:

$$\mathcal{L}_s = \mathbb{E}_{t, s_L^0, \epsilon} \left[\|s_{id} - f_\psi(\sqrt{\bar{\alpha}_t} s_L^0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, s_{mo}, t)\|^2 \right]. \quad (15)$$

Subsequently, to indirectly inject modality-guided signals into the diffusion process, we adopt a contrastive learning strategy to enhance semantic alignment across modalities. Given the denoised embeddings s_N^t and s_M^t at timestep t , the contrastive loss is defined

as \mathcal{L}_m :

$$\mathcal{L}_m = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp \left(\text{sim} \left(\mathbf{s}_{N,i}^t, \mathbf{s}_{M,i}^t \right) / \tau \right)}{\sum_{j=1}^B \exp \left(\text{sim} \left(\mathbf{s}_{N,i}^t, \mathbf{s}_{M,j}^t \right) / \tau \right)}, \quad (16)$$

where B is the batch size and τ is a temperature coefficient, and $\text{sim}(\cdot, \cdot)$ calculates cosine similarity.

4.4 Prediction and Model Optimization

The final session representation is defined as a weighted combination of the original ID-based representation and the generated neighbor representation:

$$\mathbf{s}_f = \rho \mathbf{s}_{id} + (1 - \rho) \mathbf{s}_N^0, \quad (17)$$

where $\rho \in [0, 1]$ is a learnable parameter that balances the contribution of the two components. For each candidate item $\mathbf{x}_{v_i^{id}}$, the predicted click probability \hat{y}_i is computed as:

$$\hat{y}_i = \mathbf{s}_f^\top \cdot \mathbf{x}_{v_i^{id}}. \quad (18)$$

The training objective for recommendation is defined as a standard cross-entropy loss \mathcal{L}_e , formulated as:

$$\mathcal{L}_e(y, \hat{y}) = - \sum_{i=1}^m [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)], \quad (19)$$

where y denotes the one-hot encoding vector of the ground-truth item, and \hat{y}_i is the predicted probability for the i -th item.

The overall training objective \mathcal{L} is defined as a weighted combination of the \mathcal{L}_e , \mathcal{L}_r , \mathcal{L}_s and \mathcal{L}_m :

$$\mathcal{L} = \mathcal{L}_e + \gamma(\mathcal{L}_r + \mathcal{L}_s) + \delta \mathcal{L}_m, \quad (20)$$

where γ and δ are hyperparameters that control the weights of the diffusion loss and the contrastive loss, respectively.

5 Experiment

To demonstrate the effectiveness of DiffSBR, we conduct extensive experiments guided by the following research questions:

- **RQ1:** How does DiffSBR perform compared to existing methods for SBR?
- **RQ2:** Does each proposed component contribute positively to the performance of DiffSBR?
- **RQ3:** Are the generated latent neighbors more valid than the retrieved observable neighbors from known data?
- **RQ4:** How do different hyperparameter settings influence the performance of DiffSBR?

5.1 Experimental Setup

5.1.1 Datasets and Evaluation Metrics. We evaluate our model on four widely used public datasets: **Cellphones**, **Sports**, **Grocery**, and **Instacart**, following [51]. The first three datasets are from different Amazon¹ categories and have been widely adopted in SBR. Similar to the preprocessing protocols in Zhang et al. [51, 52], sessions are constructed by grouping all user interactions that occur within a single day. The Instacart dataset is a competition dataset released on Kaggle². To simulate SBR scenarios, following Zhang

¹<http://jmcauley.ucsd.edu/data/amazon/>

²<https://www.kaggle.com/c/instacart-market-basket-analysis>

Table 1: Statistics of datasets.

Datasets	Cellphones	Sports	Grocery	Instacart
#item	9,091	14,650	7,286	10,009
#interaction	123,186	282,102	151,251	380,230
#session	40,344	90,492	43,648	88,022
avg. length	3.05	3.12	3.47	4.32

et al. [51], 20% of transactions with the shortest length are selected. Regarding modality, Amazon datasets provide both textual and visual information for each item, while for Instacart, only textual data is used. Specifically, the text modality includes item titles and brand names. Following Li et al. [17], Wu et al. [40], sessions of length 1 and items appearing fewer than five times are removed. The statistical details of all datasets are presented in Table 1. For evaluation metrics, we follow prior studies [16, 17, 51] to adopt two commonly used evaluation metrics: P@K (Precision at K) and MRR@K (Mean Reciprocal Rank at K), where $K \in \{10, 20\}$, to evaluate the performance.

5.1.2 Baselines. To demonstrate the effectiveness of DiffSBR, we compare it against a broad range of representative baselines, including traditional and state-of-the-art SBR models, and diffusion-based sequential recommendation approaches: (1) **SKNN** [13] predicts the next item based on retrieving session neighbors with high similarity from historical sessions. (2) **NARM** [17] uses a GRU with attention to model user intent. (3) **SR-GNN** [40] captures complex item transition relationships using GNN. (4) **MSGIFSR** [9] employs GNN to capture user preferences from continuous segments through co-occurrence patterns. (5) **Atten-Mixer** [48] leverages multi-level user intent to perform multi-stage reasoning on item co-occurrence transitions. (6) **MSGAT** [29] enhances the current session by retrieving neighbor based on cosine similarity and co-occurrence relationships. (7) **MGS** [16] leverages item attributes to retrieve similar neighbors and further estimate user preferences. (8) **MMSBR** [50] is the first method in SBR to combine text and images for modeling user intent. (9) **DIMO** [51] uncovers the relationships between co-occurring items and modalities to disentangle the effects of ID and modality. (10) **DiffuRec** [18] adopts diffusion models to handle sequential recommendation, replacing conventional static item embeddings with probabilistic representations. (11) **DiQDiff** [24] uses quantized user sequences as conditions to guide diffusion generation in sequential recommendation.

5.1.3 Implementation Details. Following prior studies [50, 51], we adopt the Adam optimizer with an initial learning rate of 0.001 and set the mini-batch size to 50. To ensure fair comparison, the embedding dimension for all methods is set to 100. Following Zhang et al. [51], we apply PCA to reduce both modalities to 100 dimensions. We perform grid search to select the optimal hyperparameters of the model. The number of GCN layers is set to 3, and the temperature coefficient is set to 0.3. For DDPM, we follow Mao et al. [24] by using 32 diffusion timesteps and adopting a truncated linear noise schedule.

5.2 Overall Performance (RQ1)

Table 2 reports the evaluation results of performance comparison. The results lead to several key observations:

Table 2: Comparison of different models across datasets and metrics. The best baseline results are underlined. * indicates statistically significant improvement over all baselines (p -value < 0.05).

Model	Cellphones				Sports				Grocery				Instacart			
	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20	P@10	MRR@10	P@20	MRR@20
SKNN	14.31	8.84	16.48	9.06	31.79	24.23	33.98	24.39	40.40	27.64	42.40	27.78	6.78	2.06	11.79	2.41
NARM	15.42	12.43	16.80	12.53	35.55	33.40	36.67	33.57	45.67	40.39	47.14	40.59	8.27	3.02	12.19	3.25
SR-GNN	16.36	12.96	18.11	13.09	36.31	33.36	37.69	33.66	44.33	39.44	46.24	39.64	8.96	3.27	13.00	3.64
MSGIFSR	17.80	12.40	21.16	12.64	36.27	30.36	39.65	30.59	45.45	38.16	48.15	38.35	11.56	3.74	16.44	4.02
Atten-Mixer	19.51	14.54	22.28	14.71	37.30	33.63	39.19	33.86	47.65	40.71	49.56	40.84	8.11	3.12	11.53	3.36
MSGAT	17.22	13.41	20.01	13.67	37.19	33.69	38.53	33.91	45.20	39.98	47.01	40.12	9.29	3.54	13.36	3.77
MGS	21.54	14.24	25.02	14.48	36.79	32.39	38.45	32.50	46.59	38.83	48.37	38.98	8.95	2.87	13.74	3.09
MMSBR	20.59	13.94	22.82	14.13	36.69	32.52	38.29	32.73	46.05	39.01	47.89	39.23	9.89	3.61	14.37	3.84
DIMO	<u>31.66</u>	<u>16.98</u>	<u>38.81</u>	<u>17.36</u>	<u>45.07</u>	<u>34.86</u>	<u>49.86</u>	<u>35.15</u>	<u>53.03</u>	<u>41.81</u>	<u>57.01</u>	<u>41.98</u>	<u>12.51</u>	<u>4.31</u>	<u>18.36</u>	<u>4.81</u>
DiffuRec	25.78	15.54	30.68	15.88	41.25	33.86	47.31	34.15	50.78	38.22	54.39	38.47	9.36	3.58	13.81	3.88
DiQDiff	28.12	16.41	33.19	16.83	43.01	34.51	49.28	34.96	52.29	39.04	56.03	39.32	10.62	3.96	15.34	4.17
DiffSBR	37.28*	18.03*	45.97*	18.60*	49.94*	35.66*	55.83*	36.07*	57.05*	42.64*	62.33*	42.84*	13.83*	5.12*	20.34*	5.57*
Improvement \uparrow	17.75%	6.18%	18.45%	7.14%	10.81%	2.29%	11.97%	2.62%	7.58%	1.99%	9.33%	2.05%	10.55%	18.79%	10.78%	15.80%

(1) Early methods such as SKNN rely solely on session similarity for neighbor retrieval, but lack the capacity to capture complex item transitions within sessions, resulting in limited recommendation accuracy. Later models like NARM and SR-GNN introduce attention and GNN mechanisms to enhance session modeling, thus improving performance. Recent methods such as MSGIFSR, Atten-Mixer, MSGAT, and MGS attempt to combine neighbor retrieval with graph-based modeling. By integrating historically similar sessions as auxiliary information, these models aim to enrich the session representation, and thereby improve recommendation performance. Nevertheless, they may rely only on observed neighbors and struggle to capture unobserved but semantically relevant sessions. In contrast, DiffSBR introduces a retrieval-augmented diffusion module that treats retrieved neighbors as semantic priors to guide the generation of latent neighbors, leading to a higher performance.

(2) We observed that MMSBR and DIMO achieved good performance, indicating that the multi-modal information introduced played a positive role in improving the session modeling effect. Diffusion-based models such as DiffuRec and DiQDiff also showed promising results, demonstrating the advantages of diffusion in reconstructing complex distributions and enabling condition-controlled generation. In comparison, the proposed DiffSBR further improves the recommendation performance. DiffSBR generates latent neighbors by continuously injecting both prior knowledge and multi-modal signals into the diffusion-based generation process, thereby exploring the full interest space beyond the observed data.

(3) DiffSBR consistently outperforms all baselines across multiple datasets, demonstrating its effectiveness. This performance gain can be attributed to the crucial role of latent neighbors, as well as the effectiveness of the Retrieval-augmented and Self-augmented Diffusion modules. By leveraging retrieved real neighbors as prior knowledge to guide the generation process and injecting multi-modal signals, DiffSBR is able to generate high-quality latent neighbors beyond the scope of observed data, which strengthens session representations and thereby improving recommendation accuracy.

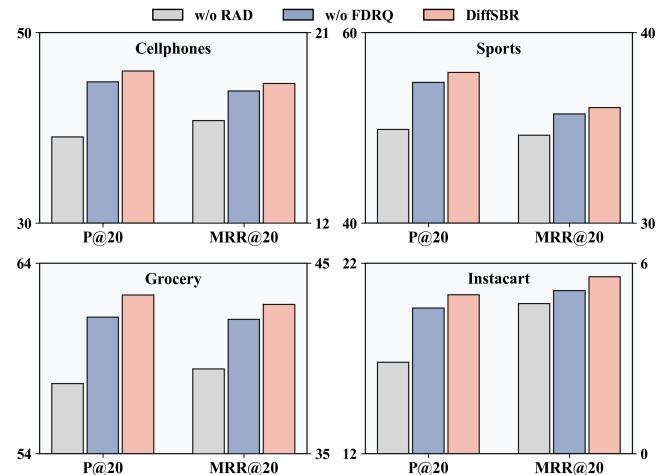


Figure 3: Effect of retrieval-augmented diffusion module.

5.3 Ablation Studies (RQ2)

To assess the impact of each major component in our method, we perform ablation studies by selectively removing or altering core modules.

5.3.1 Effect of Retrieval-augmented Diffusion Module. To verify the effect of the Retrieval-augmented Diffusion Module, we compare with variants: **w/o RAD**: This variant removes the retrieval-augmented diffusion module, meaning that no retrieved prior information is used to guide the diffusion process, nor is the retriever optimized via feedback, and no multi-modal information is incorporated. **w/o FDRQ**: In this variant, within the retrieval-augmented diffusion module, only the feedback-driven retriever optimization is removed. The performance variations across four benchmark datasets are illustrated in Figure 3.

The w/o RAD variant leads to a significant performance drop, indicating the importance of prior knowledge in guiding the diffusion process to generate semantically aligned neighbor representations. Without informative guidance, the diffusion process is more likely to deviate in latent space, resulting in degraded recommendation quality. Moreover, the w/o FDRQ variant weakens the collaboration

Table 3: Effect of self-augmented diffusion module.

Method	Cellphones		Sports		Grocery		Instacart	
	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20	P@20	MRR@20
w/o SAD	43.19	17.76	53.90	35.43	60.70	41.73	18.55	4.97
-EF	40.20	17.15	50.41	34.58	58.83	39.18	15.85	4.54
-LF	43.56	17.95	52.69	34.77	61.07	41.51	19.15	4.77
-CF	39.98	17.32	51.09	35.08	59.08	40.53	17.56	4.79
-AF	44.19	17.68	53.82	34.83	61.40	40.84	19.89	5.30
DiffSBR	45.97*	18.60*	55.83*	36.07*	62.33*	42.84*	20.34*	5.57*

between the retriever and the generator. Without dynamic supervision from the generator, the retriever cannot adaptively refine its scoring function to better support the generation process, thereby limiting the quality of retrieved guidance signals and lowering the performance.

5.3.2 Effect of Self-augmented Diffusion Module. We design these variants: **w/o SAD**: This variant removes the self-augmented diffusion module, meaning that multi-modal information is not utilized in the whole framework and only the ID modality is used. **-EF**: This variant removes the self-augmentation diffusion module and adopts an early-fusion strategy, where the modality embeddings are directly integrated into the input of the retrieval-augmented diffusion module. **-LF**: This variant removes the self-augmentation diffusion module and performs late fusion by incorporating modality embeddings at the prediction stage. **-CF**: This variant removes the self-augmentation diffusion module and performs conditional fusion by directly merging the modality embeddings into the retrieval-augmented diffusion module’s condition, which is then used to guide the diffusion denoising process. **-AF**: This variant keeps the self-augmentation diffusion module but replaces the contrastive learning with a cross-modal attention mechanism to integrate modality information.

As shown in Table 3, for the variant w/o SAD, the performance drops significantly when this variant is removed, indicating that adding multi-modal information to the diffusion process is helpful in improving the quality of latent neighbors. We further evaluate four simplified alternatives (EF, LF, CF, and AF), each integrating multi-modal information through a different fusion strategy. Although these variants incorporate multi-modal information to some extent, they all exhibit clear performance degradation compared to the full model, largely because their fusion mechanisms are either too coarse or insufficiently aligned with the diffusion denoising process. As a result, they fail to provide effective multi-modal signals, leading to weaker latent neighbor quality and reduced model performance. This result highlights that simple fusion strategies are insufficient for effectively leveraging multi-modal signals.

5.4 Effectiveness of the Generated Latent Neighbors (RQ3)

To comprehensively assess the necessity and effectiveness of latent neighbor generation, we conduct both quantitative and qualitative analyses. Specifically, we compare two variants: Observable-Neighbors, which retrieves neighbors directly from observed data based on similarity (reflecting traditional retrieval-based methods). Latent-Neighbors, which generates latent neighbors through our proposed DiffSBR approach.

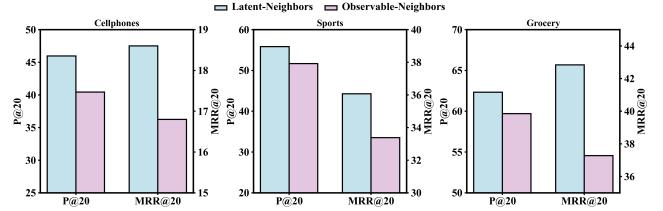


Figure 4: Comparison results of the Latent-Neighbors vs. Observable-Neighbors.

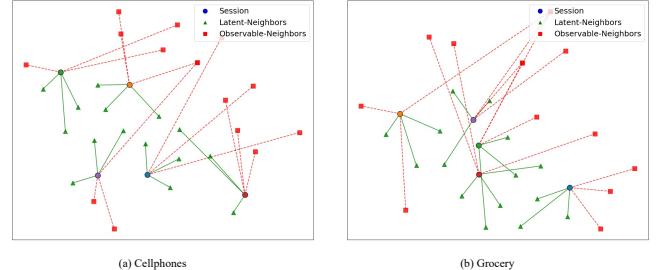


Figure 5: Qualitative visualization of the Latent-Neighbors and Observable-Neighbors.

Quantitative Evaluation. To assess the effectiveness of latent neighbor generation, we conduct systematic comparisons between two variants: Latent-Neighbors and Observable-Neighbors across three benchmark datasets. As shown in Figure 4³, Latent-Neighbors consistently outperforms Observable-Neighbors in both P@20 and MRR@20 across three datasets. These improvements highlight the advantage of our diffusion-based generation approach, which enables the synthesis of semantically relevant neighbors beyond the observed interaction data. By bridging the gaps left by static retrieval methods, latent neighbors not only enhance overall recommendation performance, but also offer data-driven evidence for the feasibility of generative neighbors modeling.

Qualitative Visualization. To intuitively understand why latent neighbors outperform static retrieval ones, we perform a visualization analysis to compare their spatial distribution. Specifically, we project the high-dimensional representations of the target sessions and their corresponding neighbors into a 2D space using t-SNE[32, 33]. We use blue dots to represent different sessions. The red squares in the figure represent the Observable-Neighbors, which are the three neighbors with the highest similarity retrieved from the known dataset. The green triangles in the figure represent the Latent-Neighbors generated using the proposed method.

As illustrated in Figure 5³, overall, latent neighbors (denoted by green triangles) exhibit a more concentrated distribution in the embedding space, closely adjacent to the target session node. In contrast, observable neighbors (depicted by red squares) display a comparatively scattered distribution, with some positioned farther from the target session. This phenomenon suggests that while conventional retrieval methods can identify relatively similar neighbors from observed data, the retrieved neighbors are often confined

³ It is worth noting that we have conducted experiments on four datasets. The observed patterns are consistent with those shown in the Figure. For the sake of visual clarity and presentation aesthetics, we display only a subset of the results, which are representative rather than accidental.

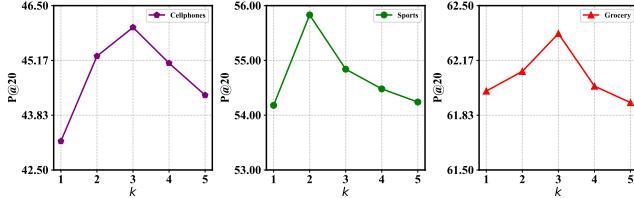


Figure 6: Impact of the number of retrieved neighbors k .

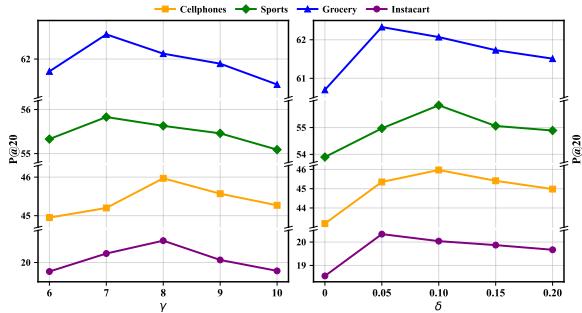


Figure 7: Impact of loss weights γ and δ .

to the scope of recorded behaviors. Conversely, latent neighbors effectively fill the sparse regions of the embedding space, thereby uncovering neighbors that are semantically closer to the target session. These qualitative results substantiate the necessity and effectiveness of our latent neighbor generation approach.

5.5 Sensitivity Analysis (RQ4)

5.5.1 Impact of the Number of Retrieved Neighbors k . The number of retrieved neighbors k , acting as prior knowledge to guide the generation of latent neighbors, serves as a key hyperparameter. To investigate its effect, we vary k in $\{1, 2, 3, 4, 5\}$ and evaluate the model performance across datasets. As shown in Figure 6, increasing k initially improves performance, as more informative neighbors provide richer semantic prior signals to the generator. However, performance peaks at $k = 2$ for the Sports dataset, at $k = 3$ for both Cellphones and Grocery, beyond which further increasing k leads to slight declines. This trend suggests that introducing too many neighbors may inject noise or redundant information, ultimately weakening the effectiveness of guidance.

5.5.2 Impact of Loss Weights γ and δ . We further investigate the impact of the two loss weights: the diffusion model loss weight γ and contrastive loss weight δ . As shown in Figure 7, for the diffusion loss weight γ , we analyze the results from $\gamma = 6$, as preliminary experiments indicated that smaller values (e.g., $\gamma < 6$) led to suboptimal performance due to insufficient influence of the diffusion loss on the generation process. Specifically, the Sports and Grocery datasets reach their peak performance at $\gamma = 7$, while the Cellphones and Instacart datasets perform best at $\gamma = 8$. This suggests that an appropriate weighting of the diffusion model loss allows the model to effectively generate latent neighbors. However, setting γ too high (e.g., $\gamma = 9$ or 10) results in a slight decline in performance, likely because the diffusion loss term dominates the overall objective and diminishes the contribution of other important components.

For the contrastive loss weight δ , performance improves with a moderate contrastive signal. Specifically, the Grocery and Instacart datasets achieve their best performance at $\delta = 0.05$, while the Cellphones and Sports datasets reach their peak at $\delta = 0.1$. This confirms the effectiveness of injecting multi-modal information during the diffusion process. However, excessively large δ values may interfere with the optimization of other loss components, leading to slightly reduced performance.

6 Conclusion

In this study, we propose a novel Diffusion-based Latent Neighbor Generation model for improving SBR. Specifically, we design a retrieval-augmented diffusion module that leverages retrieved neighbors as prior knowledge to guide the diffusion process in generating latent neighbors. Within this module, we further introduce a new training strategy to enhance the synergy between retrieval and diffusion during neighbor generation. In addition, we develop a self-augmented diffusion module to fully exploit multi-modal information throughout the diffusion process, thereby improving the quality of generated neighbors. Experimental results on four benchmark datasets demonstrate that DiffSBR consistently achieves significant performance gains over state-of-the-art methods.

In this work, we highlight the potential of latent neighbor generation for SBR. However, this remains a preliminary study on modeling latent neighbors, and there is still room for improvement. In future work, we plan to investigate the distinct strengths and complementarities of various generative models (e.g., autoregressive models) and explore hybrid frameworks that integrate multiple generative paradigms, aiming to further enhance the quality of generated latent neighbors.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (62402093), and the Sichuan Science and Technology Program (2025ZNSFSC0479).

References

- [1] Guojia An, Jing Sun, Yuhang Yang, and Fuming Sun. 2024. Enhancing collaborative information with contrastive learning for session-based recommendation. *Information Processing & Management (IPM)* 61 (2024), 103738.
- [2] Guojia An, Jie Zou, Jiwei Wei, Chaoning Zhang, Fuming Sun, and Yang Yang. 2025. Beyond whole dialogue modeling: Contextual disentanglement for conversational recommendation. In *SIGIR*, 31–41.
- [3] Xiao Ao, Shuxi Han, Yeming Li, Heli Ma, Pengfei Zhang, and Jie Zou. 2025. Retrieval Augmented Multi-agent Recommender. In *WWW*, 2968–2972.
- [4] Jinpeng Chen, Jianxiang He, Huan Li, Senzhang Wang, Yuan Cao, Kaimin Wei, Zhenye Yang, and Ye Ji. 2025. Hierarchical Intent-guided Optimization with Pluggable LLM-Driven Semantics for Session-based Recommendation. In *SIGIR*, 1655–1665.
- [5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 4171–4186.
- [6] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. In *NeurIPS*, 8780–8794.
- [7] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. 2017. Video captioning with attention-based LSTM and semantic consistency. *Transactions on Multimedia (TMM)* 19, 9 (2017), 2045–2055.
- [8] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *NeurIPS*, 1–9.
- [9] Jianan Guo, Yaming Yang, Xiangchen Song, Yuan Zhang, Yujing Wang, Jing Bai, and Yan Zhang. 2022. Learning multi-granularity consecutive user intent unit for session-based recommendation. In *WSDM*, 343–352.

[10] Muskan Gupta, Priyanka Gupta, and Lovekesh Vig. 2024. Guided Diffusion-based Counterfactual Augmentation for Robust Session-based Recommendation. *arXiv preprint arXiv:2410.21892* (2024).

[11] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *NeurIPS*. 6840–6851.

[13] Dietmar Jannach and Malte Ludewig. 2017. When recurrent neural networks meet the neighborhood for session-based recommendation. In *RecSys*. 306–310.

[14] Yangqin Jiang, Lianghao Xia, Wei Wei, Da Luo, Kangyi Lin, and Chao Huang. 2024. DiffMM: Multi-modal diffusion model for recommendation. In *MM*. 7591–7599.

[15] Yangqin Jiang, Yuhao Yang, Lianghao Xia, and Chao Huang. 2024. Diffkg: Knowledge graph diffusion model for recommendation. In *WSDM*. 313–321.

[16] Sisi Lai, Erli Meng, Fan Zhang, Chenliang Li, Bin Wang, and Aixin Sun. 2022. An attribute-driven mirror graph network for session-based recommendation. In *SIGIR*. 1674–1683.

[17] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural attentive session-based recommendation. In *CIKM*. 1419–1428.

[18] Zihao Li, Aixin Sun, and Chenliang Li. 2023. Diffurec: A diffusion model for sequential recommendation. *Transactions on Information Systems (TOIS)* 42 (2023), 1–28.

[19] Zihao Li, Xianzhi Wang, Chao Yang, Lina Yao, Julian McAuley, and Guandong Xu. 2023. Exploiting explicit and implicit item relationships for session-based recommendation. In *WSDM*. 553–561.

[20] Qidong Liu, Fan Yan, Xiangyu Zhao, Zhaocheng Du, Huifeng Guo, Ruiming Tang, and Feng Tian. 2023. Diffusion augmentation for sequential recommendation. In *CIKM*. 1576–1586.

[21] Hanzhen Lu and Zhongxin Liu. 2024. Improving Retrieval-Augmented Code Comment Generation by Retrieving for Generation. In *ICSME*. 350–362.

[22] Mingyang Lv, Xiangfeng Liu, and Yuanbo Xu. 2025. Dynamic multi-interest graph neural network for session-based recommendation. In *AAAI*. 12328–12336.

[23] Haokai Ma, Yimeng Yang, Lei Meng, Ruobing Xie, and Xiangxu Meng. 2024. Multimodal conditioned diffusion model for recommendation. In *WWW*. 1733–1740.

[24] Wenyu Mao, Shuchang Liu, Haoyang Liu, Haozhe Liu, Xiang Li, and Lantao Hu. 2025. Distinguished quantized guidance for diffusion-based sequence recommendation. In *WWW*. 425–435.

[25] Zhiqiang Pan, Fei Cai, Yanxiang Ling, and Maarten De Rijke. 2020. An intent-guided collaborative machine for session-based recommendation. In *SIGIR*. 1833–1836.

[26] Yunchen Pu, Zhe Gan, Ricardo Henao, Xin Yuan, Chunyuan Li, Andrew Stevens, and Lawrence Carin. 2016. Variational autoencoder for deep learning of images, labels and captions. In *NeurIPS*. 1–9.

[27] Shutong Qiao, Wei Zhou, Fengji Luo, and Junhao Wen. 2023. Noise-reducing graph neural network with intent-target co-action for session-based recommendation. *Information Processing & Management (IPM)* 60 (2023), 103517.

[28] Shutong Qiao, Wei Zhou, Junhao Wen, Chen Gao, Qun Luo, Peixuan Chen, and Yong Li. 2025. Multi-view Intent Learning and Alignment with Large Language Models for Session-based Recommendation. *Transactions on Information Systems (TOIS)* 43 (2025), 1–25.

[29] Shutong Qiao, Wei Zhou, Junhao Wen, Hongyu Zhang, and Min Gao. 2023. Bi-channel multiple sparse graph attention networks for session-based recommendation. In *CIKM*. 2075–2084.

[30] Ruihong Qiu, Zi Huang, Jingjing Li, and Hongzhi Yin. 2020. Exploiting cross-session information for session-based recommendation with graph neural networks. *Transactions on Information Systems (TOIS)* 38 (2020), 1–23.

[31] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *CVPR*. 1–9.

[32] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research (JMLR)* 9, 11 (2008).

[33] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. 2017. Adversarial cross-modal retrieval. In *MM*. 154–162.

[34] Wenjie Wang, Yiyuan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. 2023. Diffusion recommender model. In *SIGIR*. 832–841.

[35] Zheng Wang, Zhenwei Gao, Yang Yang, Guoqing Wang, Chengbo Jiao, and Heng Tao Shen. 2024. Geometric matching for cross-modal retrieval. *Transactions on Neural Networks and Learning Systems (TNNLS)* 36, 3 (2024), 5509–5521.

[36] Zhendong Wang, Yifan Jiang, Huangjie Zheng, Peihao Wang, Pengcheng He, Zhangyang Wang, Weizhu Chen, Mingyuan Zhou, et al. 2023. Patch diffusion: Faster and more data-efficient training of diffusion models. In *NeurIPS*. 72137–72154.

[37] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. 2020. Global context enhanced graph neural networks for session-based recommendation. In *SIGIR*. 169–178.

[38] Yibian Wei, Jie Zou, Weikang Guo, Guoqing Wang, Xing Xu, and Yang Yang. 2025. MSCRS: Multi-modal semantic graph prompt learning framework for conversational recommender systems. In *SIGIR*. 42–52.

[39] Junda Wu, Cheng-Chun Chang, Tong Yu, Zhanhui He, Jianing Wang, Yupeng Hou, and Julian McAuley. 2024. Coral: collaborative retrieval-augmented large language models improve long-tail recommendation. In *SIGKDD*. 3391–3401.

[40] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based recommendation with graph neural networks. In *AAAI*. 346–353.

[41] Zihao Wu, Xin Wang, Hong Chen, Kaidong Li, Yi Han, Lifeng Sun, and Wenwu Zhu. 2023. Diff4rec: Sequential recommendation with curriculum-scheduled diffusion augmentation. In *MM*. 9329–9335.

[42] Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui. 2021. Self-supervised graph co-training for session-based recommendation. In *CIKM*. 2180–2190.

[43] Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Lizhen Cui, and Xiangliang Zhang. 2021. Self-supervised hypergraph convolutional networks for session-based recommendation. In *AAAI*. 4503–4511.

[44] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. 2023. Diffusion models: A comprehensive survey of methods and applications. *Computing Surveys (CSUR)* 56 (2023), 1–39.

[45] Zhengyi Yang, Jiancan Wu, Zhicai Wang, Xiang Wang, Yancheng Yuan, and Xiangnan He. 2023. Generate what you prefer: Reshaping sequential recommendation via guided diffusion. In *NeurIPS*. 24247–24261.

[46] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *SIGIR*. 729–732.

[47] Feng Yu, Yanqiao Zhu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2020. TAGNN: Target attentive graph neural networks for session-based recommendation. In *SIGIR*. 1921–1924.

[48] Peiyan Zhang, Jiayan Guo, Chaozhuo Li, Yueqi Xie, Jae Boum Kim, Yan Zhang, Xing Xie, Haohan Wang, and Sunghun Kim. 2023. Efficiently leveraging multi-level user intent for session-based recommendation via atten-mixer network. In *WSDM*. 168–176.

[49] Xiaokun Zhang, Hongfei Lin, Bo Xu, Chenliang Li, Yuan Lin, Haifeng Liu, and Fenglong Ma. 2022. Dynamic intent-aware iterative denoising network for session-based recommendation. *Information Processing & Management (IPM)* 59 (2022), 102936.

[50] Xiaokun Zhang, Bo Xu, Fenglong Ma, Chenliang Li, Liang Yang, and Hongfei Lin. 2023. Beyond co-occurrence: Multi-modal session-based recommendation. *Transactions on Knowledge and Data Engineering (TKDE)* 36 (2023), 1450–1462.

[51] Xiaokun Zhang, Bo Xu, Zhaochun Ren, Xiaochen Wang, Hongfei Lin, and Fenglong Ma. 2024. Disentangling id and modality effects for session-based recommendation. In *SIGIR*. 1883–1892.

[52] Xiaokun Zhang, Bo Xu, Liang Yang, Chenliang Li, Fenglong Ma, Haifeng Liu, and Hongfei Lin. 2022. Price does matter! modeling price and interest preferences in session-based recommendation. In *SIGIR*. 1684–1693.

[53] Jujia Zhao, Wang Wenjie, Yiyuan Xu, Teng Sun, Fuli Feng, and Tat-Seng Chua. 2024. Denoising diffusion recommender model. In *SIGIR*. 1370–1379.

[54] Pan Zhiqiang, Cai Fei, Chen Wanyu, Chen Chonghao, and Chen Honghui. 2022. Collaborative graph learning for session-based recommendation. *Transactions on Information System (TOIS)* 40 (2022), 72.

[55] Jie Zou, Mohammad Aliannejadi, Evangelos Kanoulas, Shuxi Han, Heli Ma, Zheng Wang, Yang Yang, and Heng Tao Shen. 2025. PSCon: Product Search Through Conversations. In *SIGIR*. 3659–3669.

[56] Jie Zou, Yifan Chen, and Evangelos Kanoulas. 2020. Towards question-based recommender systems. In *SIGIR*. 881–890.