Information Processing and Management 63 (2026) 104318

Contents lists available at ScienceDirect

Information Processing and Management

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

Multi-Scale Transformers with dual attention and adaptive masking for sequential recommendation

Haiqin Li [a], Yuhan Yang [b], Jun Zeng [a], Min Gao [a], Junhao Wen [a,*]

[a] School of Big Data and Software Engineering, Chongqing University, Chongqing, 401331, China
[b] School of Public Administration, Sichuan University, Chengdu, 610065, China

ARTICLE INFO

Keywords:
Sequential recommendation
Mulit-scale modeling
Dual attention

ABSTRACT

Sequential recommendation focuses on modeling and predicting a user's next actions based on their sequential behavior patterns, using the temporal order and dynamics of user actions to provide more personalized and contextual suggestions. Sequential recommendation models rely on limited temporal scales, making it challenging to explicitly capture diverse user behaviors spanning multiple scales. Motivated by this challenge, this paper introduces ScaleRec, an advanced Multi-Scale Transformer architecture augmented with dual attention mechanisms and adaptive masking for sequential recommendation. ScaleRec integrates interaction granularity and context through multi-scale division, segmenting user behavior sequences into patches of varying lengths. Dual attention explicitly models fine-grained interests and coarse-grained preferences, including intra-patch cross-attention and inter-patch self-attention. Specifically, intra-patch cross-attention employs a learnable Gaussian kernel to introduce locality-based inductive biases, capturing fine-grained behavioral dynamics. The inter-patch self-attention is further enhanced by a Context-adaptive Preferences Aggregator, which dynamically selects and integrates relevant long-term user preferences. Additionally, we introduce an adaptive masking fusion strategy to filter redundant information dynamically. Extensive experiments on six benchmark datasets show that ScaleRec achieves state-of-the-art performance, improving the recommendation performance by up to 24.95% in terms of HR@5. The code of the proposed model is available at: https://github.com/gangtann/ScaleRec.

1. Introduction

Recommendation systems have attracted sustained interest due to their indispensable value across diverse application scenarios (Zhang, Yao, Sun and Tay, 2019), including e-commerce (Ge et al., 2020), media streaming (Zhang, Zhu et al., 2021), social networking, and online advertising (Zhao, Lu, Cai, He, & Zhuang, 2016). In real-world environments, user interactions exhibit pronounced temporal dynamics, underscoring the necessity of sequential recommendation (SR) methods that leverage ordered interaction sequences to predict future user interests (Boka, Niu, & Neupane, 2024; Gao et al., 2023; Qiu, Huang, Yin, & Wang, 2022).

Accurately modeling and predicting user preferences necessitates the fusion of information captured across multiple temporal scales (Du et al., 2023; Zeng, Tao, Tang, Wen, & Gao, 2025). From user behaviors that vary both individually and collectively, as well as across daily sessions and weekly or monthly cycles (Huang et al., 2019; Ren et al., 2019; Zhong, Zeng, Wang, Zhou, & Wen, 2024),

* Corresponding author.
E-mail addresses: lihaiqin@stu.cqu.cn (H. Li), yyh@stu.scu.edu.cn (Y. Yang), zengjun@cqu.edu.cn (J. Zeng), gaomin@cqu.edu.cn (M. Gao), jhwen@cqu.edu.cn (J. Wen).

https://doi.org/10.1016/j.ipm.2025.104318
Received 21 April 2025; Received in revised form 18 July 2025; Accepted 21 July 2025
Available online 31 July 2025
0306-4573/© 2025 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

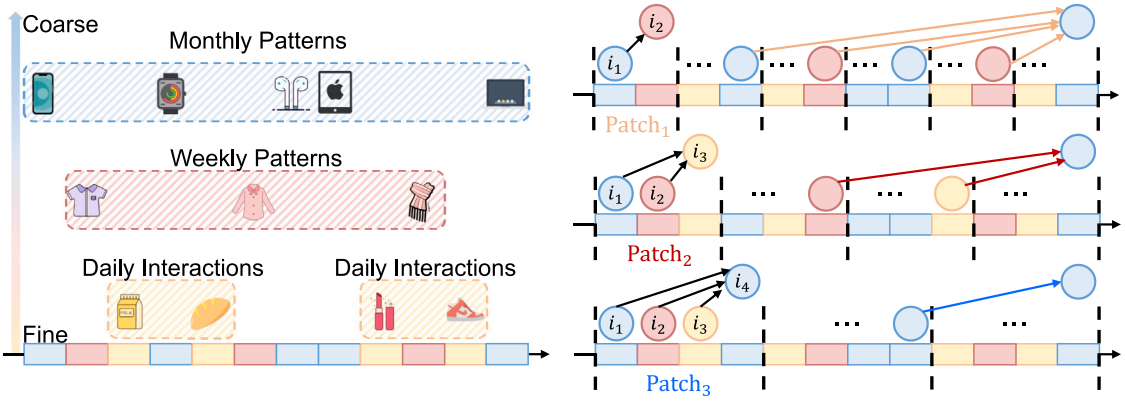


Fig. 1. Motivation for multi-scale modeling in sequential recommendation. Left: The user interaction sequence is divided into patches of varying lengths to represent different granularities, with red, orange, and blue indicating distinct patch sizes. Right: Fine-scale behavioral patterns (denoted by black arrows) and broader, global preference relationships (indicated by colored arrows) are captured by explicitly leveraging varying positional spans throughout the sequence. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

to preference dynamics reflecting interests evolving at multiple temporal horizons, models for sequential recommendation should explicitly promote multi-scale awareness representations. Transformer-based architectures have recently gained prominence across diverse research areas, demonstrating superior performance over traditional RNN-based approaches, particularly in recommendation scenarios (Chen, Liu, Li, McAuley, & Xiong, 2022; Wu et al., 2021; Xie et al., 2022). Although Transformer-based methods have recently dominated SR tasks and established benchmark-setting performance, most innovations (Li et al., 2022; Liu et al., 2023; Shin, Choi, Wi, & Park, 2024) primarily target reducing the inherent quadratic complexity or refining inductive biases – such as modifications in self-attention mechanisms or model structures – rather than explicitly addressing multi-scale representation learning. Crucial relationships capturing user preferences across multiple temporal scales are typically learned only implicitly, as existing Transformer-based sequential recommendation architectures predominantly rely on stacked attention layers and lack explicit inductive priors designed to foster cross-scale modeling. Notably, prominent models such as GRU4Rec (Hidasi, 2015), SASRec (Kang & McAuley, 2018), BERT4Rec (Sun et al., 2019), DuoRec (Qiu et al., 2022), and the recent ICSRec (Qin et al., 2024) have achieved significant results, yet none have explicitly addressed the challenging issue of multi-scale user preferences. BSARec (Shin et al., 2024) introduced scale-awareness into sequential recommendation by employing Fourier transforms to embed inductive biases capable of capturing intricate sequential dependencies. At the same time, it synergistically integrates low- and high-frequency signals within a unified framework, effectively mitigating over-smoothing in user behavior modeling. Nonetheless, this architectural inductive bias was confined strictly to a binary frequency decomposition. Considering the critical role of multi-scale characteristics in sequential recommendation, a natural question arises: can Transformer architectures be further enhanced to explicitly accommodate richer multi-scale awareness modeling?

In sequential recommendation, multi-scale modeling inherently demands attention to two key aspects: interaction granularity and context. Interaction granularity corresponds to how we segment user behavior sequences in the model and establishes the temporal scope of each modeled interaction segment. As shown on the left of Fig. 1, identical sequences can be partitioned into either smaller patches (highlighted in yellow) or larger patches (highlighted in blue), thus yielding interaction features at varying granularities. The notion of interaction context pertains to the explicit modeling of dependencies among user interactions, governing the temporal distances across interactions utilized in sequential recommendation frameworks. As shown on the right of Fig. 1, modeling the correlations between temporally proximate interactions captures local transient interests, while considering interactions separated by longer intervals captures global preference trends.

Motivation. In this work, we aim to advance the multi-scale modeling capabilities within Transformer architectures, thereby deepening our exploration into their potential for effectively capturing correlation structures in sequential recommendation tasks. The primary challenge limiting effective multi-scale modeling in Transformers is the incompleteness of multi-scale modeling. Analyzing user behaviors at distinct interaction granularities inherently shapes the scale at which sequential dependencies are modeled. However, relying solely on varying interaction sequences' granularity implicitly limits the explicit representation of dependencies across multiple interaction contexts. Explicitly accounting for diverse interaction contexts facilitates capturing dependencies spanning multiple granularities, ranging from transient user interests to enduring preference patterns. However, accurately distinguishing between fine-grained and coarse-grained interests remains inherently challenging, as it critically depends on the segmentation criteria chosen for user behavior sequences. This inherent dependency reveals a fundamental limitation: adopting a single interaction granularity viewpoint constrains the model's capacity to fully capture the multi-scale complexity of user interests.

Our Method. Motivated by insights from multi-scale modeling, this work proposes ScaleRec—a multi-scale Transformer framework integrating dual attention and adaptive masking for sequential recommendation. The multi-scale division is proposed to segment user behavior sequences into patches of distinct lengths, facilitating representations of diverse interaction granularity.

Table 1
Notations used in the paper: the left column lists each abbreviation, while the right column provides its corresponding description.

Notation	Definition
$\mathcal{U}$	The set of all users
$\mathcal{V}$	The set of all items
u	An arbitrary user, where $u \in \mathcal{U}$
v_n^u	The item with which user u interacts at the nth time step
$\lvert H^u \rvert$	The number of items with which user u interacts in the sequence
$H^u = [v_1^u, v_2^u, \dots, v_{\lvert H^u \rvert}^u]$	The sequence of items with which user u interacts, arranged in chronological order
L	The fixed maximum sequence length
d	The embedding dimension
$S = \{S_1, \dots, S_M\}$	Scale size pool, i.e., the set of candidate scale sizes
M	The total number of selectable scales in the scale size pool
K	The number of scales selected from the scale size pool
S_k	The patch length at the kth scale
P	The number of patches for a given scale
$X^p \in \mathbb{R}^{S_k \times d}$	Embedding matrix of the pth patch
σ^p	Learnable Gaussian scale parameter
G_k^p	Gaussian smoothing matrix
Z^p	Smoothed patch representation
$\text{Attn}_{\text{intra}}^p$	Intra-patch attention output for the pth patch (vector of size d_m)
$\text{Attn}_{\text{intra}}$	Aggregated intra-patch attention outputs across P patches ($\mathbb{R}^{P \times d_m}$)
X_{inter}	Inter-patch representation ($\mathbb{R}^{P \times d_m'}$)
d_m'	Feature dimension after patch aggregation
$\text{Attn}_{\text{inter}}$	Inter-patch attention output ($\mathbb{R}^{P \times d_m'}$)
$\text{Attn}_{\text{fusion}}$	Fused output of intra- and inter-patch attention
B	Binary masking matrix
$\mathcal{P}$	Masking probability
$\tilde{\text{Attn}}_{\text{fusion}}$	Masked fused output
$\text{Attn}_{\text{final}}$	Final aggregated representation for prediction task

Guided by the resulting patch lengths, we introduce a dual attention mechanism consisting of inter-patch self-attention and intra-patch cross-attention. Specifically, the former is designed to distill persistent user preferences across distinct patches, whereas the latter aims to encode fleeting user interests embedded within each individual patch. By incorporating prior knowledge of data locality, a learnable Gaussian kernel is introduced into intra-patch cross-attention to emphasize the adjacent concentration inductive bias. Moreover, we propose a Context-adaptive Preferences Aggregator that dynamically selects and integrates relevant long-term user preferences in inter-patch self-attention. We propose adaptive masking fusion by randomly masking subsequence-level patches of the interacted sequence, effectively filtering redundant or irrelevant information. Empirical evaluations on six benchmark datasets indicate that ScaleRec achieves superior recommendation performance compared to eight representative baseline approaches. Furthermore, through extensive additional analyses, we highlight the critical importance of explicitly addressing multi-scale modeling and validate the efficacy of our proposed solution, thereby demonstrating significant gains in both recommendation accuracy and model generalization.

Contributions. This paper makes the following key contributions:

- Going beyond previous methods, we tackle intricate user behavior dynamics in sequential recommendation by modeling interactions at multiple temporal scales. We leverage disentangled user behavior patterns and complementary recommendation capabilities across different scales.
- We propose a novel model, Multi-**Scale** Transformers with dual attention for Sequential **Rec**ommendation (ScaleRec), that integrates the two perspectives of the multi-scale information into both inter-patch and intra-patch, empowered by our tailored Gaussian kernel and Context-adaptive Preferences Aggregator.
- Extensive evaluation on six benchmark datasets demonstrates ScaleRec's outperformance over eight baseline methods, validating its effectiveness in improving recommendation performance.

2. Related work

2.1. Sequential recommendation

Sequential recommendation seeks to forecast the next interaction by modeling temporal patterns in users' historical behavior. Early approaches primarily relied on Markov Chain transition matrices (MC) (He & McAuley, 2016; Rendle, 2010), Frequent Pattern Mining (FPM) (Rendle, Freudenthaler, & Schmidt-Thieme, 2010), and latent factor vector models such as TranSRec (He, Kang, & McAuley, 2017), yet they struggled to capture long-term dependencies. With the latest advancements in deep learning (Hidasi, 2015; Kang & McAuley, 2018; Vaswani, 2017), researchers have increasingly focused on constructing frameworks based on Recurrent Neural Networks (RNNs) (Hidasi, 2015; Hidasi & Karatzoglou, 2018; Quadrana, Karatzoglou, Hidasi, & Cremonesi, 2017), Convolutional Neural Networks (CNNs) (Tang & Wang, 2018; Yan, Cheng, Kang, Wan, & McAuley, 2019; Yuan, Karatzoglou,

Arapakis, Jose, & He, 2019), Graph Neural Networks (GNNs) (Chang et al., 2021, 2023; Ni, Zhou, Wen, Hu, & Qiao, 2023; Qiao, Zhou, Luo, & Wen, 2023; Wu et al., 2019), or Transformers (Fan et al., 2022; Sun et al., 2019). GRU4Rec (Hidasi, 2015) and Caser (Tang & Wang, 2018) are representative sequential recommendation models based on RNNs and CNNs, respectively. SR-GNN (Wu et al., 2019) transforms session sequences into graph structures and employs GNNs to model complex item–item transition relationships. Built upon self-attention, Transformers exhibit remarkable power in modeling complex sequential relationships. Consequently, Transformers (de Souza Pereira Moreira, Rabhi, Lee, Ak, & Oldridge, 2021; Liu, Fan, Wang, & Yu, 2021) and attention mechanisms (Wu, Li, Hsieh, & Sharpnack, 2020) have been widely adopted in sequential recommendation for effectively capturing long-range dependencies in user behavior. SASRec (Kang & McAuley, 2018) is the first model to capture historical dependencies in sequences using a unidirectional Transformer, while BERT4Rec (Sun et al., 2019) advanced this by integrating bidirectional context. Compared to Transformer-based architectures, an MLP approach significantly reduces computational time complexity. MLP4Rec (Li et al., 2022) leverages an MLP-based architecture with tri-directional fusion to capture sequential, channel-wise, and feature-wise dependencies, while AutoMLP (Li, Zhang et al., 2023) is a simple, efficient, and automated approach that dynamically adjusts the short-term interest window length based on task requirements.

2.2. Self-attention mechanism in sequence recommendation

Transformer-based approaches (Zhang, Zhao et al., 2019) have progressively emerged as the dominant paradigm in sequential recommendation tasks in the past few years, primarily relying on their powerful self-attention mechanisms (Fan et al., 2022; Yue et al., 2024). Moreover, studies have begun to explore using self-supervised contrastive learning to enhance Transformer performance further (Li, Zhang, Li, Yuan, & Zhou, 2025; Zhang, Zhang, Liao, Li and Wang, 2025). Owing to its ability to generate effective auxiliary supervision signals, self-supervised learning has demonstrated outstanding performance in sequence models, helping alleviate data sparsity issues. DCRec (Yang et al., 2023) decouples user interests from herd behavior and combines contrastive learning for cross-view training. DuoRec (Qiu et al., 2022) introduces a dropout-based, model-level enhancement method that generates challenging, positive samples from the target item. ICLRec (Chen et al., 2022) employs contrastive learning to fuse intent representations with sequential models. ICSRec (Qin et al., 2024) employs coarse-grained and fine-grained contrastive learning strategies to optimize user intent representations by contrasting similar sub-sequences.

Meanwhile, some studies have focused on enhancing the computational efficiency of Transformers to reduce model complexity. For example, LightSANs (Fan, Liu et al., 2021) significantly improves computational efficiency by employing low-rank decomposition of self-attention; LinRec (Liu et al., 2023) introduces L2-normalized linear attention method that effectively preserves the advantages of the attention mechanism while reducing computational cost without compromising performance; STRec (Li, Wang et al., 2023) proposes an efficient sparse Transformer to tackle sparse attention in Transformer-based recommendation; and FMLPRec (Zhou, Yu, Zhao, & Wen, 2022) replaces the self-attention component in Transformers with learnable filters in the frequency domain, significantly reducing complexity while maintaining efficient feature extraction. In addition, a number of studies have examined the limitations of conventional self-attention mechanisms in sequential recommendation tasks. For example, AWRSR (Liu, Liu, & Liu, 2024) addresses the shortcomings in explicitly using attention weights to capture higher-order correlations; AC-TSR (Zhou et al., 2023) calibrates unreliable or inaccurate attention weight distributions; and FAME (Liu, Zhang, & Long, 2025) overcomes the limitations of conventional self-attention mechanisms in handling the multifaceted nature of items and the diversity of user preferences. However, these efforts focus primarily on enhancing self-attention and do not explicitly address the multi-scale dynamics inherent in user behavior, underscoring the need for further innovations that integrate scale-aware modeling.

2.3. Multi-scale modeling

Multi-scale feature modeling has consistently demonstrated its effectiveness in learning complex correlations and extracting robust features, as widely evidenced in the fields of CV (Fan, Xiong et al., 2021; Liu, Zhang, Zhang, Lin, & Zuo, 2018; Zhang, Dai et al., 2021), NLP (Nawrot et al., 2021), time series forecasting (Chen et al., 2023; Ding, Wu, Sun, Guo, & Guo, 2020), and multimodal learning (Hu, Singh, Darrell, & Rohrbach, 2020; Wang et al., 2022). Multi-scale Vision Transformers (Fan, Xiong et al., 2021) integrate hierarchical multi-scale representations with the Transformer architecture to significantly improve image and video recognition. However, these models primarily focus on the spatial domain and are not directly transferable to other application scenarios. In the domain of sequential recommendation, recent studies, such as BSARec (Shin et al., 2024), have utilized frequency analysis tools like the Fourier transform to map user interaction sequences into the frequency domain. This enables explicit separation and processing of high- and low-frequency signals, thereby introducing a degree of scale awareness (Du et al., 2023). However, this structural prior is limited to only two scales—low and high frequency, which means it does not capture the diverse scale information in user behavior. In contrast, our work overcomes this limitation by proposing ScaleRec, a dual-attention multi-scale Transformer for sequential recommendation that achieves comprehensive scale awareness and precisely captures the interdependencies and fine-grained characteristics among different scales. M2Rec (Zhang, Qu et al., 2025) employs FFT-based frequency decomposition combined with a Mamba architecture and LLM embeddings. Pathformer (Chen et al., 2024), on the other hand, adopts fixed-length segmentation with path-routing attention for dense time-series forecasting. In contrast, ScaleRec explicitly segments sequences into multi-scale patches and incorporates a learnable Gaussian kernel with context-adaptive attention aggregation, enabling local smoothness and long-term interest modeling, which makes it particularly suitable for sparse and discrete sequential recommendation scenarios.

Algorithm 1 Forward Pass of ScaleRec

Require: useritem sequence $H = [v_1, \dots, v_L]$, where $v_i \in \mathcal{V}$ ▷ L: sequence length
Require: scale pool $\{S_1, \dots, S_M\}$, CPA kernels $\{k_1, \dots, k_N\}$
$H_e \leftarrow \text{EmbeddingLayer}(H)$ ▷ embed each item token
for S_k **in** $\{S_1, \dots, S_M\}$ **do** ▷ iterate over scales
  $X \leftarrow \text{Segment}(H_e, S_k)$ ▷ $X \in \mathbb{R}^{P \times S_k \times d}$
  $Q^p_{\text{intra}} \leftarrow initialize$ ▷ $Q^p_{\text{intra}} \in \mathbb{R}^{1 \times d_m}$
  $K^p_{\text{intra}}, V^p_{\text{intra}} \leftarrow X^p \mathbf{W}_K, X^p \mathbf{W}_V$ ▷ $X^p \in \mathbb{R}^{S_k \times d}, \mathbf{W}_* \in \mathbb{R}^{d \times d_m}$
  $\sigma^p \leftarrow X^p \mathbf{W}_\sigma$ ▷ $\sigma^p \in \mathbb{R}^{S_k \times d_m}$
  $G^p_k = \text{Rescale}\left(\left[\frac{1}{\sqrt{2\pi}\sigma^p_i} \exp\left(-\frac{|j-i|^2}{2(\sigma^p_i)^2}\right)\right]_{i,j \in \{1,\dots,S_k\}}\right)$ ▷ $G^p_k \in \mathbb{R}^{S_k \times S_k}$
  $K^p_g, V^p_g \leftarrow G^p_k K^p_{\text{intra}}, G^p_k V^p_{\text{intra}}$ ▷ local smoothing
  $\text{Attn}^p_{\text{intra}} \leftarrow \text{Softmax}\left((Q^p_{\text{intra}} (K^p_g)^\top / \sqrt{d_m})\right) V^p_g$ ▷ $\text{Attn}^p_{\text{intra}} \in \mathbb{R}^{S_k \times d_m}$
  $\text{Attn}_{\text{intra}} \leftarrow \text{Concat}\left(\text{Attn}^1_{\text{intra}}, \dots, \text{Attn}^P_{\text{intra}}\right)$ ▷ $\text{Attn}_{\text{intra}} \in \mathbb{R}^{P \times S_k \times d_m}$
  $X_{\text{inter}} \leftarrow \text{Reshape}(\text{Linear}(X))$ ▷ $X_{\text{inter}} \in \mathbb{R}^{P \times d'_m}, d'_m = S_k \cdot d_m$
  $Q_{\text{inter}}, K_{\text{inter}}, V_{\text{inter}} \leftarrow X_{\text{inter}} \mathbf{W}_Q, X_{\text{inter}} \mathbf{W}_K, X_{\text{inter}} \mathbf{W}_V$ ▷ $Q_{\text{inter}}, K_{\text{inter}}, V_{\text{inter}} \in \mathbb{R}^{P \times d'_m}$
  $V_{\text{CPA}} \leftarrow \text{Softmax}(L(V_{\text{inter}})) \cdot \left[\text{AvgPool}_{k_1}(V_{\text{inter}}), \dots, \text{AvgPool}_{k_N}(V_{\text{inter}})\right]$ ▷ adaptive long-term preference
  $\text{Attn}_{\text{inter}} \leftarrow \text{Reshape}(\text{Softmax}(Q_{\text{inter}}(K_{\text{inter}})^\top / \sqrt{d_m}) V_{\text{CPA}})$ ▷ $\text{Attn}_{\text{inter}} \in \mathbb{R}^{P \times S_k \times d_m}$
  $\text{Attn}_{\text{fusion}} \leftarrow \text{FFN}(\text{Attn}_{\text{inter}}) + \text{FFN}(\text{Attn}_{\text{intra}})$ ▷ merge intra/inter
  $\text{Attn}_{\text{fusion}} \leftarrow \text{Linear}(\text{Reshape}(\text{Attn}_{\text{fusion}}))$ ▷ $\text{Attn}_{\text{fusion}} \in \mathbb{R}^{L \times d}$
  $B \leftarrow$ Bernoulli distribution ▷ adaptive mask
  $\tilde{\text{Attn}}_{\text{fusion}} = B \odot \text{Attn}_{\text{fusion}}$
end for
$\text{Attn}_{\text{final}} = \sum_{k=1}^{K} \tilde{\text{Attn}}^{(k)}_{\text{fusion}}$ ▷ aggregate over scales
$\hat{y} = \text{softmax}\left(\mathbf{E}^\top \text{Attn}_{\text{final}}\right)$ ▷ top-K logits
return $\hat{y}$

3. Preliminaries

Sequential recommendation aims to predict the next item a user will interact with by modeling their historical interaction sequence. Let $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ and $\mathcal{V} = \{v_1, v_2, \dots, v_{|\mathcal{V}|}\}$ denote the sets of user and item, respectively, with $|\mathcal{U}|$ and $|\mathcal{V}|$ indicating their corresponding cardinalities. For each user $u \in \mathcal{U}$, their interacted items can be arranged in chronological order as a sequence $H^u = [v^u_1, v^u_2, \dots, v^u_{|H^u|}]$. Here, $|H^u|$ is the number of the interacted items of u in the sequence. Each $v^u_n \in \mathcal{V}, 0 \le n \le |H^u|$ denoting the item interacted with at the nth step. The sequential recommendation objective is formalized as

$$\hat{v}^u_{|H^u|+1} = \arg\max_{v \in \mathcal{V}} P(v^u_{|H^u|+1} = v \mid H^u), \tag{1}$$

where $P(v \mid H^u)$ represents the probability that item v is the next interaction for user u given their past sequence H^u.

4. Proposed method

Our proposed architecture consists of four key components: (A) *Embedding Layer*, (B) *Multi-scale Division*, (C) *Dual Attention*, (D) *Adaptive Masking Fusion*. Fig. 2 illustrates how our method enables scale-awareness and multi-scale modeling. The pseudocode for the complete forward pass of ScaleRec is detailed in Algorithm 1. Key symbols and their meanings are summarised in Table 1.

4.1. Embedding layer

The embedding layer captures item properties and multi-scale user behaviors in sequential recommendation tasks. To this end, we map discrete item IDs into a continuous embedding space and explicitly incorporate positional embeddings to encode sequential order.

First, we fix the sequence length at L. If a user's interaction sequence exceeds L, we retain the latest L interactions; if shorter, we pad it with a special index (0) at the beginning to reach length L. Next, a trainable item embedding matrix $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times d}$ then maps each item ID to a d-dimensional latent vector $\mathbf{e}_t$. Additionally, a positional embedding matrix $\mathbf{P} \in \mathbb{R}^{L \times d}$ is employed to add positional biases that capture temporal order information. Finally, Layer Normalization and Dropout are applied to the combined embeddings to enhance training stability and generalization performance.

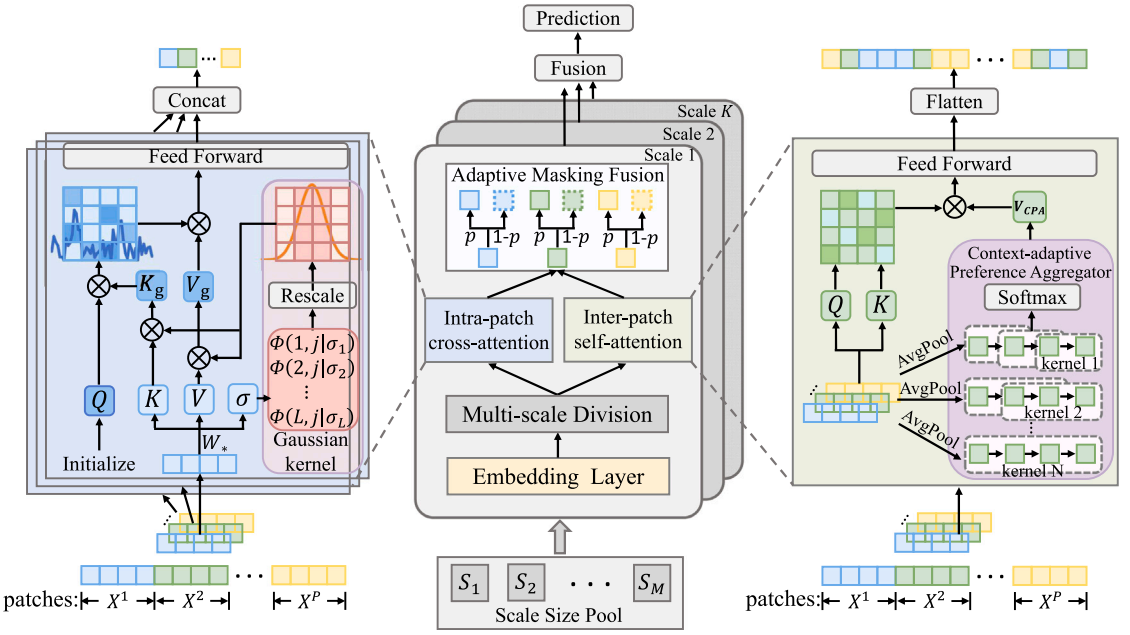


Fig. 2. Architecture of our proposed ScaleRec, which primarily comprises four core components: Multi-scale Division, Intra-patch cross-attention, Inter-patch self-attention, and an adaptive masking fusion module.

4.2. Multi-scale division

We define the scale size pool $\mathcal{S} = \{S_1, \ldots, S_M\}$ as a set of M distinct candidate scales, each corresponding to a unique segmentation of the user interaction sequence. Given a fixed sequence length L and embedding dimension d, for each scale size $S_k \in \mathcal{S}$, the original sequence is segmented into P patches, where $P = L/S_k$. Specifically, each patch $X^p \in \mathbb{R}^{S_k \times d}$ contains S_k consecutive user interactions. This multi-scale division enables parallel extraction of user interest features at different temporal scales. During training, these segmented representations are integrated using a dual attention mechanism, explicitly balancing fine-grained transient user interests and coarse-grained stable preferences. Consequently, our approach enhances the model's capability to accurately and comprehensively model entangled user behavior dynamics. This multi-scale division works synergistically with the dual attention mechanism presented in the following section, collaboratively enabling comprehensive modeling of user behaviors across multiple temporal scales.

4.3. Dual attention

Motivated by segmentations of user interactions at distinct scales, we introduce a dual attention mechanism explicitly designed to capture dependencies in user preferences across temporally partitioned interactions. Specifically, leveraging patch-based segmentation, our mechanism facilitates modeling across multiple interaction contexts at varying temporal resolutions. The proposed dual attention comprises intra-patch cross-attention, which extracts fine-grained dependencies within individual patches, and inter-patch self-attention, which models broader dependencies across patches, as illustrated in Fig. 2.

4.3.1. Intra-patch cross-attention

Motivated by the intricate multi-scale patterns inherent in user behavior sequences, we incorporate a Gaussian kernel into the intra-patch cross-attention module to explicitly model the dependencies in adjacent interaction contexts via a smoothing inductive bias. Users exhibit transient interests and stable long-term preferences, presenting complex temporal dependencies that are challenging to model by standard point-wise attention alone. To address this, we integrate a Gaussian kernel $\Phi(x, y|\sigma)$ into our intra-patch cross-attention module, introducing a smoothing inductive bias that naturally prioritizes behaviors at closer interaction context. This kernel effectively captures transient user interests while reducing noise from irrelevant distant interactions by assigning higher attention weights to temporally adjacent behaviors. Furthermore, we utilize a learnable scale parameter σ^p, allowing the attention distribution to adapt flexibly to various user behavioral patterns across multiple scales. The Gaussian prior $G_k^p \in \mathbb{R}^{d_m \times S_k \times S_k}$ is formally defined by the following expression:

$$G_k^p = \text{Rescale}\left(\left[\frac{1}{\sqrt{2\pi}\sigma_i^p}\exp\left(-\frac{|j-i|^2}{2(\sigma_i^p)^2}\right)\right]_{i,j\in\{1,\ldots,S_k\}}\right), \tag{2}$$

where Rescale(·) refers to L1 row normalization by index i. The parameter σ_i^p is a learnable scale parameter at each position i within the patch p, controlling the range of local smoothing. Specifically, σ^p is initialized through a linear projection applied directly to the input patch features, defined as:

$$\sigma^p = X^p \mathbf{W}_\sigma, \tag{3}$$

where $X^p \in \mathbb{R}^{S_k \times d}$ is the input feature matrix of the patch p, and $\mathbf{W}_\sigma \in \mathbb{R}^{d \times d_m}$ is a learnable weight matrix. Consequently, σ^p has dimensions $\mathbb{R}^{S_k \times d_m}$. Additionally, the Gaussian kernel G_k^p, which leverages σ_i^p, is defined as: $G_k^p \in \mathbb{R}^{S_k \times S_k}$, capturing local positional relationships within each patch.

Given the segmented patches $(X^1, X^2, \ldots, X^P)$ obtained with segment size S_k, we employ intra-patch cross-attention to capture correlations among interactions within the same scale. Specifically, for the pth patch $X^p \in \mathbb{R}^{S_k \times d}$, we derive the key and value matrices for attention operations by performing two trainable linear transformations, denoted as $K_{\text{intra}}^p, V_{\text{intra}}^p \in \mathbb{R}^{S_k \times d_m}$:

$$K_{\text{intra}}^p = X^p \mathbf{W}_K, \quad V_{\text{intra}}^p = X^p \mathbf{W}_V, \tag{4}$$

where $\mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times d_m}$ are learnable embedding matrices. For each patch representation K_{intra}^p and V_{intra}^p, the Gaussian kernel is applied to produce smoothed representations, highlighting dependencies among temporally adjacent interactions as follows:

$$K_g^p = G_k^p K_{\text{intra}}^p, \; V_g^p = G_k^p V_{\text{intra}}^p. \tag{5}$$

Additionally, we introduce a trainable query matrix $Q_{\text{intra}}^p \in \mathbb{R}^{1 \times d_m}$ to capture patch-specific transient interests. The intra-patch cross-attention performs the following operation:

$$\text{Attn}_{\text{intra}}^p = \text{Softmax}\left(Q_{\text{intra}}^p (K_g^p)^T / \sqrt{d_m}\right) V_g^p. \tag{6}$$

Following the intra-patch cross-attention operation, these embeddings derived from individual patches are concatenated, yielding the intra-patch representation denoted as $\text{Attn}_{\text{intra}} \in \mathbb{R}^{P \times S_k \times d_m}$, which explicitly encodes transient behavioral patterns within each segmented subsequence. The concat operation can be formulated as follows:

$$\text{Attn}_{\text{intra}} = \text{Concat}\left(\text{Attn}_{\text{intra}}^1, \ldots, \text{Attn}_{\text{intra}}^P\right). \tag{7}$$

4.3.2. Inter-patch self-attention

Inter-patch self-attention explicitly models dependencies among distinct patches, enabling the effective extraction of coarse-grained correlations underlying user preferences. It models dependencies between distant patches and leverages the broader interaction context to complement the fine-grained interests captured within individual patches.

After segmenting the user interaction sequence into patches, represented as $X \in \mathbb{R}^{P \times S_k \times d}$, we first transform the raw features from dimension d to a model-specific dimension d_m. Next, we rearrange the data by merging the patch length S_k with the transformed features, resulting in an inter-patch representation $X_{\text{inter}} \in \mathbb{R}^{P \times d_m'}$, where $d_m' = S_k \cdot d_m$. This aggregation of local context enables global dependency modeling across patches. In line with conventional self-attention formulations, we employ learnable linear transformations applied to X_{inter} to generate corresponding query, key, and value matrices, denoted respectively as $Q_{\text{inter}}, K_{\text{inter}}, V_{\text{inter}} \in \mathbb{R}^{P \times d_m'}$. Finally, we compute the inter-patch self-attention $\text{Attn}_{\text{inter}}$, which models the correlations among patches and captures coarse-grained user preference correlations in user behavior:

$$\text{Attn}_{\text{inter}} = \text{Softmax}\left(Q_{\text{inter}} (K_{\text{inter}})^T / \sqrt{d_m}\right) V_{\text{inter}}. \tag{8}$$

To better capture diverse long-term preference patterns and their influence on fine-grained interests, we introduce a novel Context-adaptive Preference Aggregator (CPA). Our key insight is that user behavior sequences embed multiple latent preference patterns at varying granularities, each exerting distinct influences on the user's immediate fine-grained interests. Therefore, identifying and selecting the most contextually relevant long-term interests becomes crucial for enhancing the recommendation quality.

Specifically, the CPA employs a set of multi-granularity kernels, each designed to extract stable long-term user preference at different granularities. Concretely, we first pre-define a candidate pool of kernel sizes that cover diverse temporal granularities, and then empirically select the optimal kernel combination for each dataset (as detailed in Section 5.4.4).

Given these selected kernels, we apply multiple average-pooling operations to the value matrix V_{inter}, obtaining diverse preference representations that reflect various long-term user behaviors. We then incorporate a context-adaptive gating mechanism $L(\cdot)$, implemented as a single-layer linear transformation followed by a softmax, to dynamically determine the importance of each granularity-specific preference with respect to the current fine-grained interaction context. The CPA can thus emphasize whichever long-term preferences are most informative for the current user. Formally, the CPA is defined as:

$$V_{\text{CPA}} = \text{Softmax}(L(V_{\text{inter}})) \cdot \left[\text{AvgPool}_{k_1}(V_{\text{inter}}), \ldots, \text{AvgPool}_{k_N}(V_{\text{inter}})\right], \tag{9}$$

where $\text{AvgPool}_{k_i}(\cdot)$ denotes the average pooling operation with the ith kernel, N represents the number of distinct granularity kernels and the softmax-weighted coefficients provide context-adaptive weights tailored precisely to the current interaction context.

Finally, the refined context-adaptive representations are integrated into the inter-patch self-attention module, dynamically selecting long-term user interests. Thus, our inter-patch self-attention mechanism is reformulated as follows:

$$\text{Attn}_{\text{inter}} = \text{Softmax}\left(\frac{Q_{\text{inter}} K_{\text{inter}}^T}{\sqrt{d_m}}\right) V_{\text{CPA}}, \tag{10}$$

where $\text{Attn}_{\text{inter}} \in \mathbb{R}^{P \times d'_m}$ denotes the attention output across patches, capturing global dependency information. Then, after a reshaping operation, we obtain the final inter-patch attention representation $\text{Attn}_{\text{inter}} \in \mathbb{R}^{P \times S_k \times d_m}$, explicitly aligning the global context back to the original patch-wise sequence format.

4.4. Adaptive masking fusion

User behavior sequences often exhibit intricate multi-scale characteristics, blending transient short-term impulses with stable long-term preferences. Such complexity presents challenges for standard attention mechanisms, which may become biased or overfit due to redundant or irrelevant information. To address this, we propose an Adaptive Masking Fusion module to enhance the robustness of feature representations after intra-patch cross-attention and inter-patch self-attention. Specifically, the adaptive masking operates directly on the combined attention output utilizing the feed-forward network, denoted as:

$$\text{Attn}_{\text{fusion}} = \text{FFN}(\text{Attn}_{\text{inter}}) + \text{FFN}(\text{Attn}_{\text{intra}}). \tag{11}$$

where $\text{Attn}_{\text{fusion}} \in \mathbb{R}^{P \times S_k \times d_m}$ is the fused attention representation, which integrates intra- and inter-patch contextual information. Subsequently, a reshaping operation and linear projection transform $\text{Attn}_{\text{fusion}}$ into $\mathbb{R}^{L \times d}$, restoring the original sequence dimension for final recommendation.

A binary masking matrix $B \in \{0, 1\}^{L \times d}$ is generated from a Bernoulli distribution with probability $\mathcal{P}$, selectively masking features in the fused representation. Formally, the masked fused representation is computed as:

$$\tilde{\text{Attn}}_{\text{fusion}} = B \odot \text{Attn}_{\text{fusion}}, \tag{12}$$

where $\tilde{\text{Attn}}_{\text{fusion}}$ represents the masked feature representations passed to subsequent layers. This adaptive masking enables the model to better capture essential contextual information across multiple temporal scales.

4.5. Model prediction and learning

By integrating attention outputs across multiple scales via element-wise summation, our method comprehensively captures diverse user behavior patterns, thereby enriching the final user representation. Specifically, the final attention output is computed as:

$$\text{Attn}_{\text{final}} = \sum_{k=1}^{K} \tilde{\text{Attn}}_{\text{fusion}}^{(k)}, \tag{13}$$

where K is the number of scales selected from the scale size pool $S = \{S_1, S_2, \ldots, S_M\}$ (with $K \leq M$), and $\text{Attn}_{\text{fusion}}^{(k)}$ represents the fused attention output at the kth selected scale.

We now proceed to item prediction after obtaining this comprehensive multi-scale representation of user behavior. Specifically, we derive the recommendation probabilities by multiplying $\text{Attn}_{\text{final}}$ with the trainable item embedding matrix $\mathbf{E} \in \mathbb{R}^{|\mathcal{V}| \times d}$ and applying the *softmax* function:

$$\hat{y} = \text{softmax}\left(\mathbf{E}^\top \text{Attn}_{\text{final}}\right), \tag{14}$$

where $\hat{y} \in \mathbb{R}^{|\mathcal{V}|}$ denotes the predicted scores for all items. Ideally, the ground-truth item i should receive a high score $\hat{y}_i$. During training, we employ the cross-entropy loss $\mathcal{L}_{\text{Rec}}$ as the optimization objective to enhance the model's discriminative power for true interactions:

$$\mathcal{L}_{\text{Rec}} = -\sum_{i=1}^{|\mathcal{V}|} \left[y_i \log\left(\hat{y}_i\right) + \left(1 - y_i\right) \log\left(1 - \hat{y}_i\right) \right]. \tag{15}$$

5. Experiments

This section first outlines the experimental design and setup, and then conducts a comprehensive empirical evaluation of ScaleRec based on the following research questions:

- **RQ1**: How does ScaleRec perform compared to state-of-the-art sequential recommendation methods?
- **RQ2**: What is the individual contribution of each key component to ScaleRec's performance?
- **RQ3**: How do ScaleRec's hyperparameter settings affect its effectiveness?
- **RQ4**: Can the performance improvements of ScaleRec be adequately explained?
- **RQ5**: How efficient is ScaleRec compared with baselines?

Table 2
Overview of dataset characteristics following preprocessing.

Statistics	Beauty	Sports	Toys	Yelp	LastFM	ML-1M
# Users	22 363	35 598	19 412	30 431	1090	6041
# Items	12 101	18 357	11 924	20 033	3646	3417
# Interactions	198 502	296 337	167 597	316 354	52 551	999 611
Avg. Actions/User	8.9	8.3	8.6	10.4	48.2	165.5
Avg. Actions/Item	16.4	16.1	14.0	15.8	14.4	292.6
Sparsity	99.93%	99.95%	99.93%	99.95%	98.68%	95.16%

5.1. Experimental settings

5.1.1. Experimental datasets

To comprehensively assess the model's performance under different sequence lengths and densities, we employ six publicly available recommendation datasets spanning both short- and long-sequence scenarios:

- **Beauty, Sports, and Toys**: These datasets record users' historical purchase records of various products (McAuley, Targett, Shi, & Van Den Hengel, 2015). They are segmented by product category, each exhibiting relatively short interaction sequences. We conducted experiments on three representative subcategories—Beauty, Sports, and Toys.
- **Yelp**: A large-scale dataset comprising user reviews of restaurants and bars, reflecting consumption patterns in local service recommendation.
- **LastFM**: This dataset contains user interaction data with music sourced from Last.fm music streaming platform. It records users' listening histories, ratings, and tags, making it suitable for long-sequence music recommendation tasks.
- **ML-1M**: A widely used dataset from MovieLens, characterized by the longest average interaction sequence length among the datasets used in this study (Harper & Konstan, 2015).

Based on previous studies (Xie et al., 2022; Zhou et al., 2020), we filtered out sequences shorter than five interactions and rare items occurring fewer than five times, and truncated all sequences to a maximum length of 50. For evaluation, we adopt the leave-one-out protocol (Ren et al., 2020; Sun et al., 2019), designating each user's most recent interaction as the test instance, the second most recent as the validation instance, and the remaining interactions for training. Table 2 presents the statistics of the six datasets employed in our experiments.

5.1.2. Evaluation protocols

Model performance is evaluated using Hit Rate (HR@K) and Normalized Discounted Cumulative Gain (NDCG@K) for $K \in \{5, 10, 20\}$, computed over the full item ranking without negative sampling to eliminate sampling bias (Cai, Wu, San, Wang, & Wang, 2021; Krichene & Rendle, 2020). The metrics are defined as:

$$\mathrm{HR@K} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \delta\big(\hat{R}(u) \cap R(u) \neq \varnothing\big), \tag{16}$$

$$\mathrm{NDCG@K} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \frac{\sum_{i=1}^{K} \delta\big(i \in R(u)\big) \cdot \frac{1}{\log_2(i+1)}}{\sum_{i=1}^{\min\left(|R(u)|, K\right)} \frac{1}{\log_2(i+1)}}, \tag{17}$$

where $\delta(\cdot)$ is the indicator function, $\hat{R}(u)$ denotes the set of top-K items recommended to user u, and $R(u)$ is the set of ground-truth relevant items for u.

5.1.3. Compared baseline models

To benchmark our approach, we benchmark it against several cutting-edge models for sequential recommendation, which are broadly categorized into four groups:

RNN or CNN-based Sequential Models.

- **GRU4Rec** (Hidasi, 2015): The pioneering deep learning model that first applies gated recurrent units (GRU) within an RNN architecture to model user behavior sequences.
- **Caser** (Tang & Wang, 2018): The first to introduce CNN into sequential recommendation, utilizing horizontal and vertical convolutions to model sequences.

Transformer-based Sequential Models.

- **SASRec** (Kang & McAuley, 2018): The model is the first to introduce a Transformer-based architecture with self-attention for sequential recommendation, effectively capturing user interaction patterns and yielding substantial performance gains.
- **BERT4Rec** (Sun et al., 2019): Builds upon a bidirectional Transformer with a Cloze task to provide more comprehensive contextual representations and sequence modeling.

Transformer-based Sequential Models with Contrastive Learning.

- **DuoRec** (Qiu et al., 2022): The model leverages dropout-induced stochasticity to generate diverse sequence views for unsupervised contrastive learning and then applies supervised contrastive learning to sequences sharing the same target item.
- **ICSRec** (Qin et al., 2024): This method models users' latent intentions through coarse-grained and fine-grained contrastive learning based on subsequences with the same target items and clustered intention prototypes.

Transformer-Based Frequency-Domain Sequential Models.

- **FMLPRec** (Zhou et al., 2022): This model employs learnable frequency-domain filters to denoise sequential data and capture low- and high-frequency patterns, outperforming Transformer-based methods with lower complexity.
- **BSARec** (Shin et al., 2024): It injects frequency-domain inductive bias via the Fourier transform and integrates both low- and high-frequency information, effectively alleviating the over-smoothing problem of self-attention.

5.1.4. Implementation details

All baseline models are implemented using their official open-source code and configured with the hyperparameters recommended in the original publications. For our model, the Transformer is configured with three layers and one attention head, utilizing a training batch size set to 256. We set $d = 64$. Optimization is performed with Adam at a learning rate of 1×10^{-3}, and a dropout rate of 0.5 is applied. Since each dataset has a different average sequence length, we set the number of candidate scale sizes to $M = 4$, and accordingly, the scale size pool is defined as $\{50, 25, 10, 5\}$. The adaptive masking probability $\mathcal{P}$ ranges from 0 to 1. In all experiments, the candidate kernel pool is fixed as $\{2, 3, 4, 6, 8, 12, 24, 32\}$. We adopt an early stopping criterion: training is halted if NDCG@20 fails to improve over 10 consecutive epochs, and the model with the best validation performance is retained. All reported results are now based on ten independent runs with different random seeds. All experiments are conducted on Ubuntu 22.04.1 LTS, with software environments including PyTorch 1.8.1, NumPy 1.24.3, SciPy 1.11.1, CUDA 12.6, Python 3.9.7, and NVIDIA Driver 560.45.03, on a hardware platform equipped with an Intel i6 CPU and an NVIDIA RTX 3090 GPU.

5.2. Overall performance comparison (RQ1)

Table 3 summarizes the performance comparison of our proposed method against eight competitive baseline models across six benchmark datasets. We adopt standard evaluation metrics, including HR@{5,10,20} and NDCG@{5,10,20}, to comprehensively assess recommendation effectiveness.

The results show that Transformer-based methods outperform traditional RNN- and CNN-based approaches (e.g., GRU4Rec and Caser), demonstrating the superiority of attention mechanisms for sequential recommendation tasks. Furthermore, contrastive learning–based models perform substantially better than their non-contrastive counterparts. In particular, ICSRec and DuoRec leverage contrastive objectives to learn more discriminative user and item embeddings, validating the effectiveness of contrastive learning in enhancing representation quality. Additionally, frequency-domain methods such as BSARec, which explicitly introduce inductive biases to fuse high- and low-frequency signals, have achieved substantial performance gains. Additionally, experiments on the ML-1M dataset, which features significantly longer user sequences, demonstrate that ScaleRec substantially outperforms all baselines, especially surpassing BSARec with only two scales. This result indicates that ScaleRec's multi-scale modeling is particularly effective when dealing with longer user histories, as it can better capture intricate temporal dependencies such as weekly, monthly, or seasonal user patterns. This underscores the effectiveness of incorporating frequency-domain information to enhance sequential recommendation models.

The proposed model performs best across nearly all datasets and metrics, significantly outperforming the strongest baselines, BSARec, and ICSRec. Specifically, relative improvements of up to 4.10%, 9.68%, 1.87%, 11.69%, 24.95% and 2.14% are observed on Beauty, Sports, Toys, Yelp, LastFM and ML-1M datasets, respectively. Notably, compared with methods that only consider limited or fixed scales, such as BSARec and FMLPRec, our approach further enhances performance by effectively capturing multi-scale user behavior patterns through the dual-attention framework. These results strongly validate our multi-scale dual-attention framework, demonstrating its effectiveness in addressing the challenges posed by diverse user behavior patterns.

5.3. Ablation study (RQ2)

5.3.1. Ablation study on core model components

To evaluate the individual contribution of each component, we construct six model variants—each omitting a distinct module—and perform systematic ablation experiments accordingly:

- **w/o Mask**: removes the adaptive masking fusion module, which filters redundant information dynamically;
- **w/o Scale**: removes multi-scale division, maintaining a single-scale representation;
- **w/o CPA**: removes the Context-Adaptive Preference Aggregator, which dynamically selects and integrates relevant long-term user preferences;
- **w/o Guass**: removes the Gaussian kernel, which introduces locality-based inductive biases, capturing fine-grained behavioral dynamics;
- **w/o Inter**: removes inter-scale self-attention mechanism, which captures stable user preferences across patches;

Table 3
Performance comparison of various approaches across six benchmark datasets. The best and second-best results are highlighted in bold and underlined, respectively. 'Improv.' denotes the relative performance improvement compared to the strongest baseline and all improvements over the best baseline are statistically significant ($p < 0.05$, paired t-test).

Dataset	Metric	Caser	GRU4Rec	SASRec	BERT4Rec	FMLPRec	DuoRec	ICSRec	BSARec	Ours	Improv.
Beauty	HR@5	.0125	.0169	.0340	.0469	.0346	.0707	.0698	.0707	**.0736 ± .0011**	4.10%
	HR@10	.0225	.0304	.0531	.0705	.0559	.0965	.0960	.0978	**.1003 ± .0009**	2.56%
	HR@20	.0403	.0527	.0832	.1073	.0869	.1313	.1298	.1345	**.1375 ± .0004**	2.23%
	NDCG@5	.0076	.0104	.0221	.0311	.0222	.0501	.0494	.0503	**.0509 ± .0003**	1.19%
	NDCG@10	.0108	.0147	.0283	.0387	.0291	.0584	.0579	.0590	**.0594 ± .0005**	0.68%
	NDCG@20	.0153	.0203	.0356	.0480	.0369	.0671	.0663	.0682	**.0688 ± .0004**	0.88%
Sports	HR@5	.0091	.0118	.0188	.0275	.0220	.0396	.0403	.0400	**.0442 ± .0007**	9.68%
	HR@10	.0163	.0187	.0298	.0428	.0336	.0569	.0565	.0583	**.0630 ± .0005**	8.06%
	HR@20	.0260	.0303	.0459	.0649	.0525	.0791	.0794	.0830	**.0897 ± .0005**	8.07%
	NDCG@5	.0056	.0079	.0124	.0180	.0146	.0276	.0283	.0280	**.0306 ± .0008**	8.13%
	NDCG@10	.0080	.0101	.0159	.0229	.0183	.0331	.0335	.0339	**.0367 ± .0006**	8.26%
	NDCG@20	.0104	.0131	.0200	.0284	.0231	.0387	.0393	.0401	**.0434 ± .0002**	8.23%
Toys	HR@5	.0095	.0121	.0440	.0412	.0432	.0770	.0788	.0792	**.0803 ± .0006**	1.39%
	HR@10	.0161	.0211	.0652	.0635	.0671	.1034	.1055	.1066	**.1074 ± .0003**	0.75%
	HR@20	.0268	.0348	.0929	.0939	.0974	.1369	.1368	.1405	**.1431 ± .0002**	1.86%
	NDCG@5	.0058	.0077	.0297	.0282	.0288	.0568	.0571	.0574	**.0583 ± .0003**	1.57%
	NDCG@10	.0079	.0106	.0366	.0353	.0365	.0653	.0657	.0662	**.0671 ± .0004**	1.36%
	NDCG@20	.0106	.0140	.0435	.0430	.0441	.0737	.0736	.0747	**.0761 ± .0001**	1.87%
Yelp	HR@5	.0117	.0130	.0149	.0256	.0159	.0271	.0272	.0252	**.0290 ± .0002**	6.62%
	HR@10	.0197	.0221	.0249	.0433	.0287	.0442	.0445	.0432	**.0497 ± .0008**	11.69%
	HR@20	.0337	.0383	.0424	.0717	.0490	.0717	.0715	.0704	**.0779 ± .0003**	8.65%
	NDCG@5	.0070	.0080	.0091	.0159	.0100	.0170	.0175	.0159	**.0180 ± .0003**	2.86%
	NDCG@10	.0096	.0109	.0123	.0216	.0142	.0225	.0230	.0217	**.0246 ± .0002**	6.96%
	NDCG@20	.0131	.0150	.0167	.0287	.0192	.0294	.0298	.0285	**.0316 ± .0002**	6.04%
LastFM	HR@5	.0303	.0312	.0413	.0294	.0367	.0431	.0477	.0477	**.0596 ± .0009**	24.95%
	HR@10	.0431	.0404	.0633	.0459	.0560	.0624	.0716	.0633	**.0761 ± .0012**	6.28%
	HR@20	.0642	.0541	.0927	.0596	.0826	.0963	.1083	.1046	**.1092 ± .0011**	0.83%
	NDCG@5	.0227	.0217	.0284	.0198	.0243	.0300	.0329	.0327	**.0402 ± .0008**	22.19%
	NDCG@10	.0268	.0245	.0355	.0252	.0306	.0361	.0405	.0378	**.0455 ± .0011**	12.35%
	NDCG@20	.0321	.0280	.0429	.0286	.0372	.0446	.0496	.0484	**.0537 ± .0007**	8.27%
ML-1M	HR@5	.0927	.1005	.1374	.1512	.1316	.1838	.2445	.1874	**.2485 ± .0018**	1.64%
	HR@10	.1556	.1657	.2137	.2346	.2065	.2704	.3368	.2679	**.3399 ± .0007**	0.92%
	HR@20	.2488	.2664	.3245	.3440	.3137	.3738	.4518	.3772	**.4527 ± .0021**	0.20%
	NDCG@5	.0592	.0619	.0873	.1021	.0846	.1252	.1710	.1264	**.1745 ± .0025**	2.05%
	NDCG@10	.0795	.0828	.1116	.1289	.1087	.1530	.2007	.1524	**.2050 ± .0013**	2.14%
	NDCG@20	.1028	.1081	.1395	.1564	.1356	.1790	.2297	.1800	**.2312 ± .0028**	0.65%

- **w/o Intra**: removes the intra-scale cross-attention mechanism, which captures transient user interests within individual patches.

As shown in Table 4, our complete model (ScaleRec) consistently outperforms all ablated variants, confirming the effectiveness and necessity of each designed component. Specifically, removing the Gaussian kernel (**w/o Guass**) or the intra-scale cross-attention module (**w/o Intra**) significantly degrades the recommendation performance, highlighting the importance of capturing adjacent-context patterns and short-term user interests. The **w/o Inter** variant also exhibits decreased accuracy, verifying the necessity of inter-scale self-attention in capturing persistent user preferences across distinct patches. Furthermore, the performance drop in the absence of adaptive masking (**w/o Mask**) demonstrates its effectiveness in enhancing the model's generalization by preventing overfitting to redundant patterns. The variants **w/o Scale** and **w/o CPA** similarly underperform, underscoring the contribution of multi-scale representation and Context-Adaptive Preference Aggregator to the comprehensive modeling of user behavior dynamics. In summary, the ablation study confirms that all designed components collectively contribute to the superior performance of our ScaleRec, with intra-patch cross-attention and inter-patch self-attention being particularly critical.

5.3.2. Gaussian kernel ablation and locality mechanism comparison

In this subsection, we aim to thoroughly assess the effectiveness and necessity of the proposed Gaussian kernel within the intra-patch attention mechanism. Moreover, we compare the Gaussian kernel against several alternative locality-based attention mechanisms to provide deeper insights into our design choice. Specifically, we introduce four variants for comparison, including removing the kernel entirely (denoted as None), replacing the Gaussian kernel with a polynomial projection kernel (HiPPO), a fixed-window attention mechanism (Fixed), a dilated (skipped) attention mechanism (Dilated), and a convolutional attention mechanism (Conv). The motivations behind selecting these alternatives are as follows: (1) HiPPO (Gu, Goel, & Ré, 2022) aims to project historical information onto orthogonal polynomial bases, capturing long-term continuity in contrast to Gaussian locality; (2) Fixed employs a pre-defined local attention window, explicitly enforcing a hard locality constraint; (3) Dilated sparsely samples distant

Table 4
Ablation study of ScaleRec based on HR@5 and NDCG@5 metrics across the Beauty, Sports, Toys, Yelp, and LastFM datasets.

Model	Beauty		Sports		Toys		Yelp		LastFM	
	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5	HR@5	NDCG@5
ScaleRec	**0.0736**	**0.0509**	**0.0442**	**0.0306**	**0.0803**	**0.0583**	**0.0290**	**0.0180**	**0.0596**	**0.0402**
w/o Mask	0.0715	0.0505	0.0406	0.0281	0.0763	0.0558	0.0264	0.0170	0.0541	0.0363
w/o Scale	0.0702	0.0499	0.0423	0.0296	0.0783	0.0558	0.0276	0.0177	0.0459	0.0341
w/o CPA	0.0719	0.0499	0.0433	0.0298	0.0797	0.0578	0.0284	0.0178	0.0532	0.0344
w/o Guass	0.0697	0.0499	0.0411	0.0292	0.0787	0.0582	0.0271	0.0170	0.0541	0.0356
w/o Inter	0.0712	0.0501	0.0417	0.0288	0.0761	0.0546	0.0286	0.0178	0.0486	0.0327
w/o Intra	0.0698	0.0502	0.0401	0.0283	0.0772	0.0568	0.0281	0.0179	0.0450	0.0315

Table 5
Performance comparison of locality-based attention mechanisms including Gaussian kernel.

Attention type	Beauty						Sports					
	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20
Gauss	**0.0736**	**0.1003**	**0.1375**	**0.0509**	**0.0594**	**0.0688**	**0.0442**	**0.0630**	**0.0897**	**0.0306**	**0.0367**	**0.0434**
HiPPO	0.0700	0.0973	0.1330	0.0508	0.0592	0.0686	0.0390	0.0547	0.0806	0.0273	0.0323	0.0389
Fixed	0.0710	0.0988	0.1359	0.0497	0.0587	0.0681	0.0398	0.0589	0.0849	0.0278	0.0339	0.0405
Dilated	0.0686	0.0957	0.1325	0.0492	0.0579	0.0672	0.0416	0.0594	0.0842	0.0290	0.0347	0.0409
Conv	0.0678	0.0944	0.1280	0.0488	0.0573	0.0658	0.0387	0.0563	0.0777	0.0269	0.0326	0.0379
None	0.0697	0.0954	0.1282	0.0499	0.0578	0.0661	0.0411	0.0583	0.0815	0.0292	0.0342	0.0400

interactions, expanding the receptive field without increasing computational complexity significantly; and (4) Conv pre-aggregates adjacent tokens using depth-wise convolutions, emphasizing local pattern extraction.

Table 5 presents the performance comparison of these variants across Beauty and Sports datasets using standard evaluation metrics HR and NDCG at various cutoff thresholds. First, we observe a clear performance degradation when entirely removing the Gaussian kernel ("None"), confirming its critical role in capturing fine-grained user behaviors. The Gaussian kernel specifically imposes a locality bias that smoothly emphasizes neighboring interactions while rapidly attenuating distant interactions, effectively modeling short-term and fine-grained behaviors that significantly contribute to recommendation accuracy.

Regarding alternative locality mechanisms, each variant demonstrates a distinct performance profile corresponding to its design intuition. The HiPPO kernel, optimized for capturing long-range dependencies via polynomial projection, shows comparatively modest performance, likely due to its reduced sensitivity to immediate local contexts. The Fixed-window mechanism explicitly restricts attention to a predefined local range, limiting adaptability to varying sequence contexts. Similarly, Dilated attention focuses on sparsely sampled distant interactions, sacrificing fine-grained context information that proves critical in recommendation tasks. Meanwhile, the Conv variant aggregates local interactions using convolution, achieving somewhat better performance than the aforementioned fixed or dilated mechanisms, yet still falling short compared to the Gaussian approach.

Among all mechanisms evaluated, our proposed Gaussian kernel consistently achieves superior results on both datasets, with particularly notable improvements in HR@5, HR@10, and HR@20. This indicates the Gaussian kernel's unique capability in adaptively balancing local contextual information and maintaining sensitivity to fine-grained user interaction signals.

In summary, these results clearly justify our selection of the Gaussian kernel over other locality-based mechanisms, highlighting its effectiveness in enhancing fine-grained behavioral modeling and demonstrating a favorable balance between computational efficiency and recommendation accuracy.

5.4. Hyper-parameter sensitivity analysis (RQ3)

5.4.1. Impact of the number of scale sizes

In the experiment, we set the scale pool to {50, 25, 10, 5}. We tested different combinations of $K = 1$ to $K = 4$ under the condition of other hyperparameters being optimal, with each K value corresponding to the best-performing combination for each scale. Table 6 presents the impact of different K values on recommendation performance. The results show that $K = 2$ outperforms $K = 1$ on most evaluation metrics, with $K = 2$ and $K = 3$ exhibiting relatively small differences, highlighting the advantage of modeling key multi-scale features to improve recommendation performance. Furthermore, different datasets benefit distinctly from modeling user interests and preferences across varying scales. However, it is observed that not all scale combinations produce equally effective representations; we explore this issue in further detail in Section 5.4.3. This indicates that appropriately selecting scale combinations is crucial for enhancing the performance of recommendation systems.

5.4.2. Impact of adaptive masking probability

We investigate the impact of adaptive masking probability on model performance by experimenting with different linear probability-based adaptive masking strategies. As shown in Fig. 3, different adaptive masking probabilities (i.e., $\mathcal{P}$ values) have a significant effect on model's performance on the Beauty and Sports datasets, exhibiting an overall trend of better performance in the mid-range and poorer performance at both extremes. Specifically, lower masking probabilities tend to preserve more information,

Table 6
Performance comparison for different values of K on beauty and sports datasets.

Dataset	Beauty						Sports					
Metric	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20
K = 1	0.0702	0.0978	0.1348	0.0499	0.0588	0.0682	0.0423	0.0615	0.0876	0.0296	0.0358	0.0424
K = 2	**0.0736**	**0.1003**	**0.1375**	**0.0509**	0.0594	**0.0688**	**0.0442**	**0.0630**	**0.0897**	**0.0306**	**0.0367**	**0.0434**
K = 3	0.0707	0.0986	0.1363	0.0500	0.0590	0.0685	0.0428	0.0618	0.0864	0.0296	0.0357	0.0419
K = 4	0.0717	0.0997	0.1356	0.0508	**0.0598**	**0.0688**	0.0427	0.0610	0.0856	0.0298	0.0356	0.0418

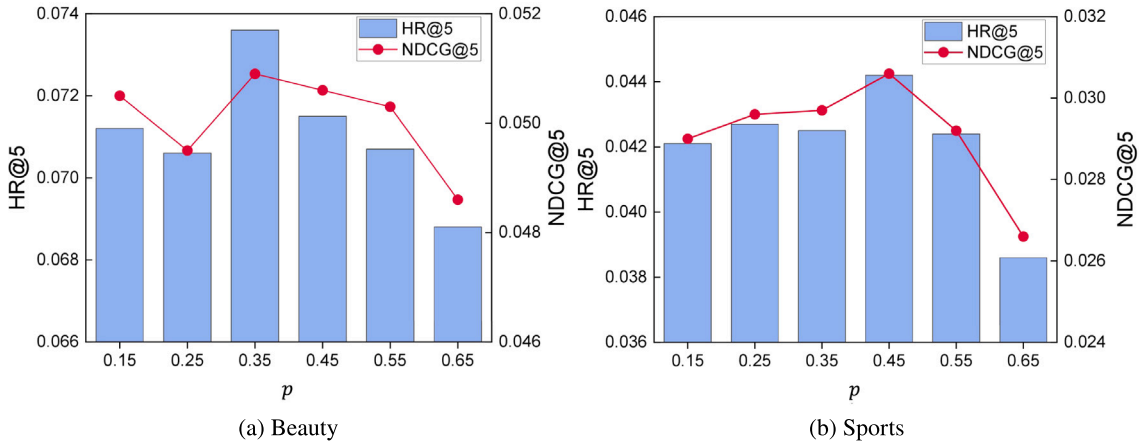


Fig. 3. Performance comparison for different values of $\mathcal{P}$ on Beauty and Sports datasets.

but they also retain excessive noise and redundant features, which may lead to overfitting; in contrast, higher dropout probabilities might result in the loss of critical information. The intermediate range of masking probabilities achieves an optimal balance between noise suppression and effective feature retention, enhancing the robustness of multi-scale representation fusion and the model's generalization ability. These findings suggest that an appropriately chosen adaptive masking strategy is crucial in promoting effective interaction among multi-scale information and capturing both short-term fluctuations and long-term behavioral patterns, ultimately leading to improved overall recommendation performance.

5.4.3. Impact of the combination of scale sizes

We examine the sensitivity regarding the combination of patch lengths used for multi-scale modeling. We select six representative scale combinations (e.g., [50,25], [50,10], [50,5], [25,10], [25,5], and [10,5]) and measure their impact on recommendation performance using HR@5 and NDCG@5 metrics on the Beauty and Sports datasets.

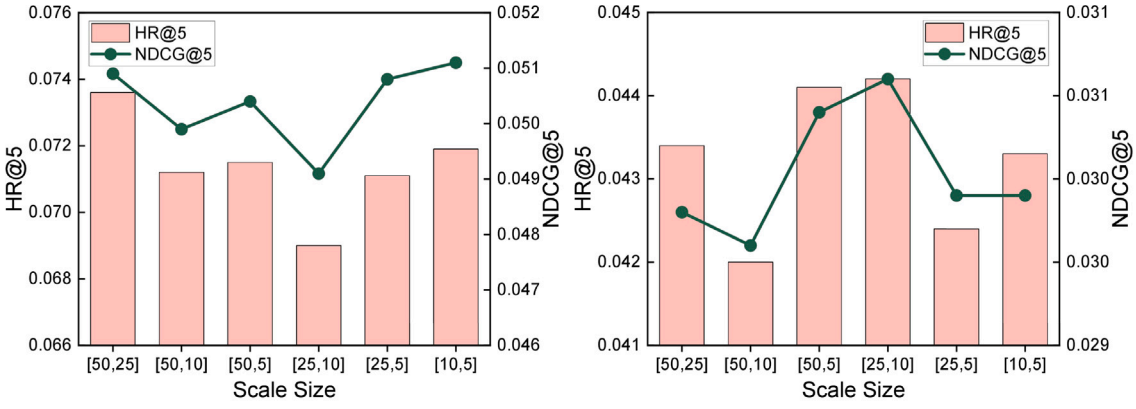
As illustrated in Fig. 4, the recommendation performance demonstrates noticeable fluctuations across different scale combinations. For the Beauty dataset, coupling the maximal window length 50 with a mid-range window 25 delivers the highest HR@5 and NDCG@5. The 50-length patch preserves virtually the entire interaction history, which is essential for uncovering the periodic repurchase cycles common in Beauty, while the 25-length patch smooths local noise and captures medium-term context. By contrast, Sports sequences are appreciably shorter and more fragmented; here the [25, 10] pairing proves most effective. A 25-length patch already spans most user histories and provides a stable view of overall interest, whereas a 10-length patch concentrates on the latest actions, thus tracking the rapid, session-level shifts that typify Sports consumption patterns.

Taken together, these results indicate that datasets characterized by longer, periodically repeating sequences favor a wide-plus-medium scale pairing (e.g., [50, 25]), whereas those with shorter, fast-changing histories benefit from a medium-plus-short configuration (e.g., [25, 10]). The analysis demonstrates that our model maintains reasonable robustness across varying scales, achieving stable and superior performance through balanced multi-scale representation learning.

5.4.4. Impact of multi-kernel sizes in CPA

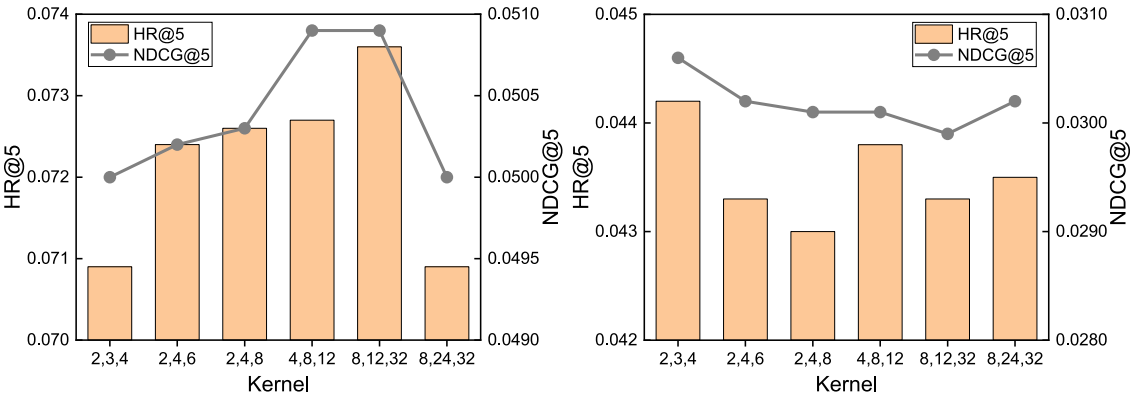
We further investigate the influence of kernel sizes within the Context-adaptive Preferences Aggregator on recommendation performance. The multi-kernel design aims to select and integrate relevant long-term user preferences dynamically. Specifically, we test multiple kernel size combinations (e.g., "2, 3, 4" denotes the simultaneous use of three kernels with sizes of 2, 3, and 4) to examine their effectiveness in capturing preferences at different granularities. The evaluation metrics used are HR@5 and NDCG@5 on the Beauty and Sports datasets.

The results presented in Fig. 5 show significant model performance differences across kernel configurations. The Beauty dataset achieves optimal performance using kernel sizes of "8, 12, 32", suggesting that combining medium to large kernels effectively captures longer-term user preferences. Conversely, for the Sports dataset, smaller kernel sizes ("2, 3, 4") yield superior results,



(a) Beauty (b) Sports

Fig. 4. Performance comparison of the six scale combinations for $K = 2$.



(a) Beauty (b) Sports

Fig. 5. Performance comparison for different kernels on Beauty and Sports datasets.

indicating that finer kernels better capture the rapid temporal dynamics characteristic of this dataset. The significant performance drops observed when employing larger kernels ("8, 24, 32") on both datasets further underscore the importance of appropriately selecting kernel sizes to match dataset-specific behavior patterns.

Overall, these findings highlight the multi-kernel strategy's benefits for effectively modeling complex sequential patterns. The appropriate selection of kernel sizes enables the model to adapt robustly to dataset-specific temporal dynamics, confirming the utility and generalizability of our CPA.

5.5. Effectiveness of adaptive masking (RQ3)

This section provides empirical evidence on the superiority of adaptive masking over the commonly adopted dropout strategy in Transformer-based sequential recommendation models. We evaluated four configurations: dropout only, adaptive masking with a fixed probability, adaptive masking with a linearly increasing masking probability based on item position in the user interaction sequence (with probabilities increasing from 0.2 up to 0.5 on the Beauty dataset, and from 0.3 up to 0.6 on the Sports dataset); and a combination of both methods. As shown in Fig. 6, all adaptive masking variants outperform dropout in both recommendation accuracy and convergence speed. Moreover, employing a sequence-position-dependent linear increase in masking probability yields a modest but consistent performance gain over fixed probability masking or its combination with dropout. These findings indicate that dynamically adjusting the masking probability during multi-scale feature fusion enables the model to more effectively attend to salient information, thereby substantially improving training efficiency and predictive performance.

5.6. Case study (RQ4)

To answer RQ4, we conducted a case study using the Beauty dataset to examine the internal mechanisms of ScaleRec's dual-attention module by visualizing the attention weights of the most recent 15 positions in user sequences. The results, shown in Fig. 7,

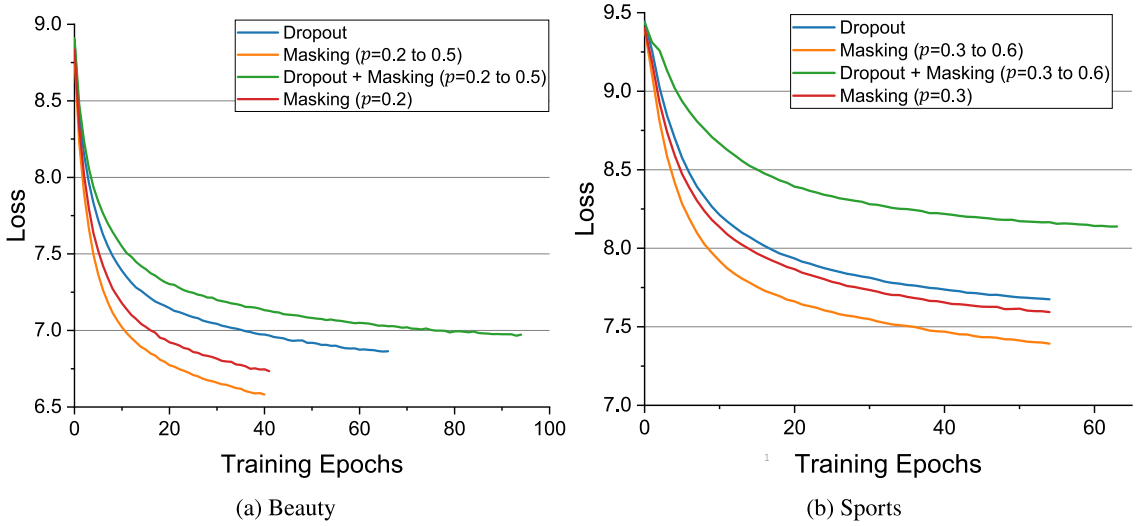


Fig. 6. Comparison of performance with adaptive masking with two different probabilities, dropout, and using both adaptive masking and dropout.

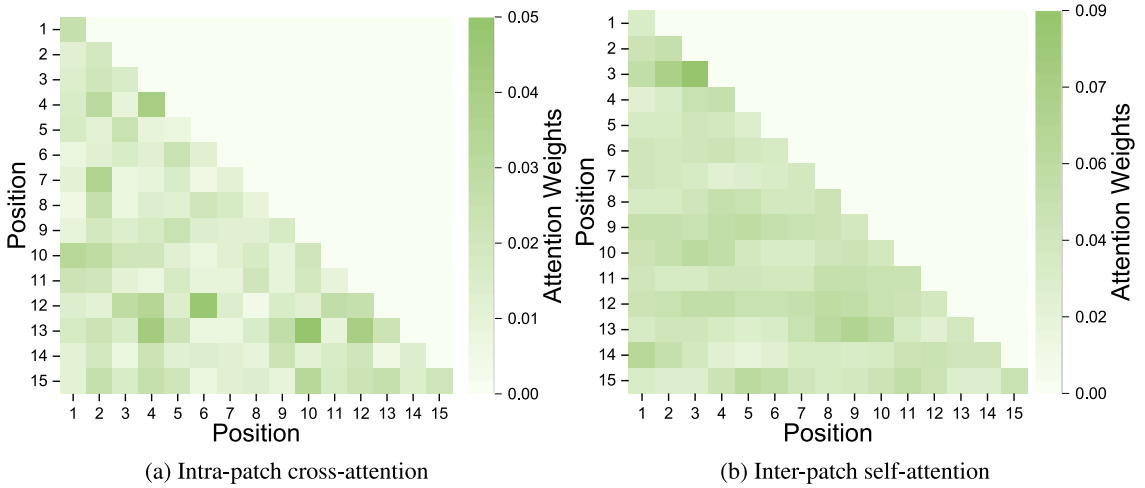


Fig. 7. Visualization of ScaleRec's dual-attention mechanism on the Beauty dataset.

reveal several key observations. the intra-patch cross-attention heatmap reveals distinct local hotspots, indicating that this module effectively captures transient fine-grained user interests, thereby enhancing the prediction of recent preferences. In contrast, the inter-patch attention visualization shows a uniform distribution, suggesting its advantage in integrating long-term dependencies and capturing stable coarse-grained user preferences across patches. These findings demonstrate that ScaleRec can simultaneously capture local, transient interests and global, stable trends through its dual-attention mechanism, achieving precise multi-scale modeling.

5.7. Efficiency comparison with baseline methods (RQ5)

To assess ScaleRec's practical efficiency for recommendation scenarios, we conduct comparative experiments measuring key computational metrics, including parameter count, training time per epoch, and inference latency over the full test set. These metrics are vital since recommendation systems require a balance between predictive accuracy and computational responsiveness.

Table 7 summarizes efficiency comparisons on the Beauty and Sports datasets. On both datasets, ScaleRec exhibits parameter counts slightly higher than the baseline models. Regarding training speed, ScaleRec consistently outperforms more computationally intensive methods such as ICSRec and DuoRec, with training times closer to lightweight approaches like SASRec and BSARec. In inference latency, ScaleRec maintains competitive performance, incurring only marginally higher latency compared to the fastest baselines but remaining within practical limits.

Table 7
Comparison of model efficiency on beauty and sports datasets.

Methods	Beauty			Sports		
	# params	Train time (s/epoch)	Inference latency (s/ total test set)	# params	Train time (s/epoch)	Inference latency (s/total test set)
ScaleRec	929,445	20.35	9.03	1,328,854	20.83	19.03
BSARec	878,208	15.85	7.14	1,278,592	26.25	19.10
ICSRec	877,888	72.43	7.86	1,278,272	105.47	14.81
DuoRec	877,824	24.99	7.45	1,278,208	48.40	15.75
SASRec	877,824	14.69	6.96	1,278,208	22.06	14.26

Overall, ScaleRec demonstrates a favorable trade-off, providing state-of-the-art recommendation accuracy without incurring significant computational overhead. These results confirm its suitability for practical deployment scenarios requiring efficient yet highly accurate recommendation models.

6. Conclusions and future work

This study proposed ScaleRec, an innovative Transformer architecture with multi-scale modeling capabilities specifically designed for sequential recommendation tasks. Unlike existing models that implicitly handle scale diversity, ScaleRec explicitly integrates multiple interaction granularities and contexts to better capture user behaviors across various temporal scales. Our proposed dual attention mechanism, complemented by the learnable Gaussian kernel and Context-adaptive Preferences Aggregator, effectively models transient fine-grained interests and stable coarse-grained preferences. The adaptive masking fusion further improves robustness by selectively filtering redundant interactions. Empirical results on six benchmark datasets validate ScaleRec's effectiveness, consistently outperforming competitive baselines and demonstrating superior generalization ability.

Nevertheless, our current approach relies on predefined patch-size combinations, which might not optimally generalize to datasets with highly variable temporal dynamics. A promising future direction involves exploring automated and adaptive selection methods for multi-scale divisions to improve performance across diverse recommendation scenarios. Additionally, incorporating side information, such as user profiles or item attributes, into our multi-scale modeling framework presents a promising research direction. Our work highlights the importance of explicitly capturing multi-scale user behavior patterns and offers valuable insights for future sequential recommendation studies.

CRediT authorship contribution statement

Haiqin Li: Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Yuhan Yang:** Writing – original draft, Methodology, Data curation, Conceptualization. **Jun Zeng:** Validation, Supervision, Project administration. **Min Gao:** Visualization, Validation, Supervision. **Junhao Wen:** Validation, Supervision, Software.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (Grant No. 62072060), and supported by the Natural Science Foundation of Chongqing, China, No. CSTB2024NSCQ-MSX0617.

Data availability

The data that has been used is confidential.

References

Boka, T. F., Niu, Z., & Neupane, R. B. (2024). A survey of sequential recommendation systems: Techniques, evaluation, and future directions. *Information Systems*, Article 102427.

Cai, R., Wu, J., San, A., Wang, C., & Wang, H. (2021). Category-aware collaborative sequential recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 388–397).

Chang, J., Gao, C., Zheng, Y., Hui, Y., Niu, Y., Song, Y., et al. (2021). Sequential recommendation with graph neural networks. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 378–387).

Chang, Y., Zhou, W., Cai, H., Fan, W., Hu, L., & Wen, J. (2023). Meta-relation assisted knowledge-aware coupled graph neural network for recommendation. *Information Processing & Management*, *60*(3), Article 103353.

Chen, L., Chen, D., Shang, Z., Wu, B., Zheng, C., Wen, B., et al. (2023). Multi-scale adaptive graph neural network for multivariate time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, *35*(10), 10748–10761.

Chen, Y., Liu, Z., Li, J., McAuley, J., & Xiong, C. (2022). Intent contrastive learning for sequential recommendation. In *Proceedings of the ACM web conference 2022* (pp. 2172–2182).

Chen, P., Zhang, Y., Cheng, Y., Shu, Y., Wang, Y., Wen, Q., et al. (2024). Pathformer: Multi-scale transformers with adaptive pathways for time series forecasting. arXiv preprint arXiv:2402.05956.

de Souza Pereira Moreira, G., Rabhi, S., Lee, J. M., Ak, R., & Oldridge, E. (2021). Transformers4rec: Bridging the gap between nlp and sequential/session-based recommendation. In *Proceedings of the 15th ACM conference on recommender systems* (pp. 143–153).
Ding, Q., Wu, S., Sun, H., Guo, J., & Guo, J. (2020). Hierarchical multi-scale Gaussian transformer for stock movement prediction.. In *IJCAI* (pp. 4640–4646).
Du, X., Yuan, H., Zhao, P., Qu, J., Zhuang, F., Liu, G., et al. (2023). Frequency enhanced hybrid attention network for sequential recommendation. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval* (pp. 78–88).
Fan, X., Liu, Z., Lian, J., Zhao, W. X., Xie, X., & Wen, J.-R. (2021). Lighter and better: low-rank decomposed self-attention networks for next-item recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 1733–1737).
Fan, Z., Liu, Z., Wang, Y., Wang, A., Nazari, Z., Zheng, L., et al. (2022). Sequential recommendation via stochastic self-attention. In *Proceedings of the ACM web conference 2022* (pp. 2036–2047).
Fan, H., Xiong, B., Mangalam, K., Li, Y., Yan, Z., Malik, J., et al. (2021). Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 6824–6835).
Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., et al. (2023). A survey of graph neural networks for recommender systems: Challenges, methods, and directions. *ACM Transactions on Recommender Systems*, *1*(1), 1–51.
Ge, Y., Zhao, S., Zhou, H., Pei, C., Sun, F., Ou, W., et al. (2020). Understanding echo chambers in e-commerce recommender systems. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 2261–2270).
Gu, A., Goel, K., & Ré, C. (2022). Efficiently modeling long sequences with structured state spaces. In *The international conference on learning representations (ICLR)*.
Harper, F. M., & Konstan, J. A. (2015). The movielens datasets: History and context. *Acm Transactions on Interactive Intelligent Systems (Tiis)*, *5*(4), 1–19.
He, R., Kang, W.-C., & McAuley, J. (2017). Translation-based recommendation. In *Proceedings of the eleventh ACM conference on recommender systems* (pp. 161–169).
He, R., & McAuley, J. (2016). Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th international conference on data mining* (pp. 191–200). IEEE.
Hidasi, B. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint arXiv:1511.06939.
Hidasi, B., & Karatzoglou, A. (2018). Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM international conference on information and knowledge management* (pp. 843–852).
Hu, R., Singh, A., Darrell, T., & Rohrbach, M. (2020). Iterative answer prediction with pointer-augmented multimodal transformers for textvqa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9992–10002).
Huang, C., Wu, X., Zhang, X., Zhang, C., Zhao, J., Yin, D., et al. (2019). Online purchase prediction via multi-scale modeling of behavior dynamics. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 2613–2622).
Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining* (pp. 197–206). IEEE.
Krichene, W., & Rendle, S. (2020). On sampled metrics for item recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 1748–1757).
Li, C., Wang, Y., Liu, Q., Zhao, X., Wang, W., Wang, Y., et al. (2023). STRec: Sparse transformer for sequential recommendations. In *Proceedings of the 17th ACM conference on recommender systems* (pp. 101–111).
Li, K., Zhang, Y., Li, X., Yuan, M., & Zhou, W. (2025). Mask diffusion-based contrastive learning for knowledge-aware recommendation. *IEEE Transactions on Knowledge and Data Engineering*.
Li, M., Zhang, Z., Zhao, X., Wang, W., Zhao, M., Wu, R., et al. (2023). Automlp: Automated mlp for sequential recommendations. In *Proceedings of the ACM web conference 2023* (pp. 1190–1198).
Li, M., Zhao, X., Lyu, C., Zhao, M., Wu, R., & Guo, R. (2022). MLP4Rec: A pure MLP architecture for sequential recommendations. arXiv preprint arXiv:2204.11510.
Liu, L., Cai, L., Zhang, C., Zhao, X., Gao, J., Wang, W., et al. (2023). Linrec: Linear attention mechanism for long-term sequential recommender systems. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval* (pp. 289–299).
Liu, Z., Fan, Z., Wang, Y., & Yu, P. S. (2021). Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 1608–1612).
Liu, Y., Liu, M., & Liu, X. (2024). Pay attention to attention for sequential recommendation. In *Proceedings of the 18th ACM conference on recommender systems* (pp. 890–895).
Liu, M., Zhang, S., & Long, C. (2025). Facet-aware multi-head mixture-of-experts model for sequential recommendation. In *Proceedings of the eighteenth ACM international conference on web search and data mining* (pp. 127–135).
Liu, P., Zhang, H., Zhang, K., Lin, L., & Zuo, W. (2018). Multi-level wavelet-CNN for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 773–782).
McAuley, J., Targett, C., Shi, Q., & Van Den Hengel, A. (2015). Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval* (pp. 43–52).
Nawrot, P., Tworkowski, S., Tyrolski, M., Kaiser, Ł., Wu, Y., Szegedy, C., et al. (2021). Hierarchical transformers are more efficient language models. arXiv preprint arXiv:2110.13711.
Ni, S., Zhou, W., Wen, J., Hu, L., & Qiao, S. (2023). Enhancing sequential recommendation with contrastive generative adversarial network. *Information Processing & Management*, *60*(3), Article 103331.
Qiao, S., Zhou, W., Luo, F., & Wen, J. (2023). Noise-reducing graph neural network with intent-target co-action for session-based recommendation. *Information Processing & Management*, *60*(6), Article 103517.
Qin, X., Yuan, H., Zhao, P., Liu, G., Zhuang, F., & Sheng, V. S. (2024). Intent contrastive learning with cross subsequences for sequential recommendation. In *Proceedings of the 17th ACM international conference on web search and data mining* (pp. 548–556).
Qiu, R., Huang, Z., Yin, H., & Wang, Z. (2022). Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining* (pp. 813–823).
Quadrana, M., Karatzoglou, A., Hidasi, B., & Cremonesi, P. (2017). Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the eleventh ACM conference on recommender systems* (pp. 130–137).
Ren, R., Liu, Z., Li, Y., Zhao, W. X., Wang, H., Ding, B., et al. (2020). Sequential recommendation with self-attentive multi-adversarial network. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval* (pp. 89–98).
Ren, K., Qin, J., Fang, Y., Zhang, W., Zheng, L., Bian, W., et al. (2019). Lifelong sequential modeling with personalized memorization for user response prediction. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval* (pp. 565–574).
Rendle, S. (2010). Factorization machines. In *2010 IEEE international conference on data mining* (pp. 995–1000). IEEE.
Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on world wide web* (pp. 811–820).
Shin, Y., Choi, J., Wi, H., & Park, N. (2024). An attentive inductive bias for sequential recommendation beyond the self-attention. *Vol. 38*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 8984–8992).
Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., et al. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 1441–1450).
Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining* (pp. 565–573).

Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.

Wang, J., Wu, Z., Ouyang, W., Han, X., Chen, J., Jiang, Y.-G., et al. (2022). M2tr: Multi-modal multi-scale transformers for deepfake detection. In *Proceedings of the 2022 international conference on multimedia retrieval* (pp. 615–623).

Wu, L., Li, S., Hsieh, C.-J., & Sharpnack, J. (2020). SSE-PT: Sequential recommendation via personalized transformer. In *Proceedings of the 14th ACM conference on recommender systems* (pp. 328–337).

Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., & Tan, T. (2019). Session-based recommendation with graph neural networks. *Vol. 33*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 346–353).

Wu, J., Wang, X., Feng, F., He, X., Chen, L., Lian, J., et al. (2021). Self-supervised graph learning for recommendation. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 726–735).

Xie, X., Sun, F., Liu, Z., Wu, S., Gao, J., Zhang, J., et al. (2022). Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering* (pp. 1259–1273). IEEE.

Yan, A., Cheng, S., Kang, W.-C., Wan, M., & McAuley, J. (2019). CosRec: 2D convolutional neural networks for sequential recommendation. In *Proceedings of the 28th ACM international conference on information and knowledge management* (pp. 2173–2176).

Yang, Y., Huang, C., Xia, L., Huang, C., Luo, D., & Lin, K. (2023). Debiased contrastive learning for sequential recommendation. In *Proceedings of the ACM web conference 2023* (pp. 1063–1073).

Yuan, F., Karatzoglou, A., Arapakis, I., Jose, J. M., & He, X. (2019). A simple convolutional generative network for next item recommendation. In *Proceedings of the twelfth ACM international conference on web search and data mining* (pp. 582–590).

Yue, Z., Wang, Y., He, Z., Zeng, H., McAuley, J., & Wang, D. (2024). Linear recurrent units for sequential recommendation. In *Proceedings of the 17th ACM international conference on web search and data mining* (pp. 930–938).

Zeng, J., Tao, H., Tang, H., Wen, J., & Gao, M. (2025). Global and local hypergraph learning method with semantic enhancement for POI recommendation. *Information Processing & Management*, *62*(1), Article 103868.

Zhang, P., Dai, X., Yang, J., Xiao, B., Yuan, L., Zhang, L., et al. (2021). Multi-scale vision longformer: A new vision transformer for high-resolution image encoding. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 2998–3008).

Zhang, Q., Qu, L., Wen, H., Huang, D., Yiu, S.-M., Hung, N. Q. V., et al. (2025). M2Rec: Multi-scale mamba for efficient sequential recommendation. arXiv preprint arXiv:2505.04445.

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, *52*(1), 1–38.

Zhang, Y., Zhang, Y., Liao, W., Li, X., & Wang, X. (2025). Multi-view self-supervised learning on heterogeneous graphs for recommendation. *Applied Soft Computing*, *174*, Article 113056.

Zhang, T., Zhao, P., Liu, Y., Sheng, V. S., Xu, J., Wang, D., et al. (2019). Feature-level deeper self-attention network for sequential recommendation.. In *IJCAI* (pp. 4320–4326).

Zhang, J., Zhu, Y., Liu, Q., Wu, S., Wang, S., & Wang, L. (2021). Mining latent structures for multimedia recommendation. In *Proceedings of the 29th ACM international conference on multimedia* (pp. 3872–3880).

Zhao, Z., Lu, H., Cai, D., He, X., & Zhuang, Y. (2016). User preference learning for online social recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *28*(9), 2522–2534.

Zhong, L., Zeng, J., Wang, Z., Zhou, W., & Wen, J. (2024). SCFL: Spatio-temporal consistency federated learning for next poi recommendation. *Information Processing & Management*, *61*(6), Article 103852.

Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., et al. (2020). S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM international conference on information & knowledge management* (pp. 1893–1902).

Zhou, P., Ye, Q., Xie, Y., Gao, J., Wang, S., Kim, J. B., et al. (2023). Attention calibration for transformer-based sequential recommendation. In *Proceedings of the 32nd ACM international conference on information and knowledge management* (pp. 3595–3605).

Zhou, K., Yu, H., Zhao, W. X., & Wen, J.-R. (2022). Filter-enhanced MLP is all you need for sequential recommendation. In *Proceedings of the ACM web conference 2022* (pp. 2388–2399).