RCMIRec: Reconstructing Dynamic Routing for Multi-Interest Sequential Recommendation

Ziyang Li[1,2], Jiwei Qin[1,2,*], Jie Ma[1,2], Shishen Li[1,2]
[1]School of Computer Science and Technology, Xinjiang University, Urumqi 830046, China
[2]Xinjiang Key Laboratory of Signal Detection and Processing, Xinjiang University, Urumqi 830046, China
107552304088@stu.xju.edu.cn, jwqin@xju.edu.cn, jmacs27@stu.xju.edu.cn, 107552303950@stu.xju.edu.cn

***Abstract*—Multi-interest sequential recommendation models employ dynamic routing mechanisms to capture diverse user interests for personalized recommendation. However, existing methods often suffer from weight over-concentration during interest learning, where a small number of highly similar items dominate each interest representation, while other relevant items are underweighted. This results in interest representations that overly rely on a few items and lack robustness. To address this issue, we propose reconstructing the dynamic routing mechanism for multi-interest sequential recommendation (RCMIRec). Specifically, we introduce a covariance routing regularization to encourage a more balanced distribution of routing weights during training. While this regularization alleviates weight over-concentration, it does not explicitly ensure accurate item–interest assignment. Therefore, we further design a backward-flow reconstruction mechanism to constrain the matching relationship between items and interests, ensuring that each item is consistently aligned with its most relevant interest representation. Extensive experiments on three public benchmark datasets demonstrate that the proposed method significantly outperforms state-of-the-art baselines.**

***Index Terms*—sequential recommendation, multi-interest learning, covariance routing regularization, dynamic routing mechanism**

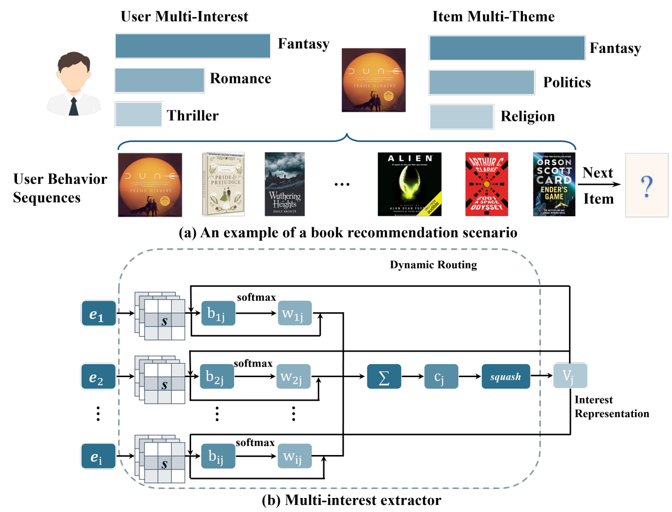

Fig. 1. (a) An example of a book recommendation scenario. (b) Dynamic routing mechanism in multi-interest sequential recommendation.

I. Introduction

Sequential recommendation, an important branch of recommender systems, aims to model user preferences by leveraging sequential dependencies and contextual information in user behavior sequences to predict users' future preferences [1]. Owing to its effectiveness in mitigating information overload, sequential recommendation has been widely adopted in e-commerce platforms and various online services [2]. Fig. 1(a) illustrates a book recommendation scenario, where users often display interests in multiple categories simultaneously, such as science fiction, thriller, and romance. Additionally, individual items typically have multiple thematic attributes. For example, *Dune* can be classified as science fiction, political, and religious literature. In such complex scenarios, where both user behaviors and item semantics are multifaceted, most existing sequential recommendation methods use a single latent vector to represent user preferences, which is insufficient to capture the full diversity and complexity of user interests [3].

To address this limitation, multi-interest learning methods [5], [6] have been introduced into the field of sequential recommendation. These methods aim to extract multiple latent interests from user behavior sequences and model each interest representation separately, thereby enabling more accurate personalized recommendation. Existing multi-interest sequential recommendation methods can generally be categorized into two types. The first type [7], [8] leverages external auxiliary information, such as item categories, to guide the multi-interest learning process. However, such methods heavily rely on high-quality and task-relevant auxiliary information, which is not always available in real-world applications. The second type [9]–[11] does not rely on external knowledge and typically adopts capsule-network-based dynamic routing mechanisms (as shown in Fig. 1(b)) to capture high-order dependencies in user interaction sequences for modeling multiple latent user interests.

Although existing multi-interest sequential recommendation models have achieved notable performance improvements, we observe that they still suffer from inherent limitations when modeling multiple interests via dynamic routing. Specifically, during the iterative routing process, routing weights tend to become overly concentrated on a small number of items that are highly similar to a given interest, causing these items to dominate the interest extraction process (as shown in Fig. 2). Consequently, when learning a particular interest representation, the model assigns disproportionately large weights to a

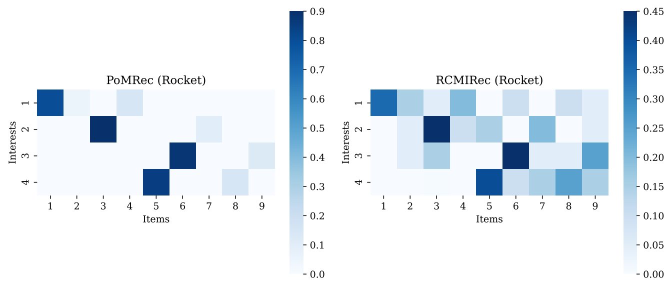

Fig. 2. Visualization of an example of item-interest routing weights of PoMRec and RCMIRec.

few highly similar items while paying insufficient attention to other relevant items with lower similarity, or even ignoring them entirely. This biased weight allocation undermines the stability, robustness, and generalization of interest representations, making them resemble memorization of individual items rather than abstract representations of user interest patterns.

To address these issues, we propose Reconstructing Dynamic Routing for Multi-Interest Sequential Recommendation (RCMIRec), which reconstructs the dynamic routing mechanism to more robustly and accurately capture users' diverse interests. First, we construct a covariance matrix over routing weights across different interests and introduce a covariance routing regularization term to constrain the routing distributions of each interest. This regularization encourages a more balanced allocation of routing weights, suppressing excessive concentration on a small number of items and enabling each interest representation to integrate information from multiple relevant items, thereby improving the stability and generalization of learned interests. However, although the regularization strategy alleviates routing-weight over-concentration, it does not explicitly guarantee that each item is assigned to its most relevant interest representation. To further strengthen item–interest alignment, we design a backward-flow reconstruction mechanism that constrains the matching relationship between items and interests during routing-weight adjustment, ensuring that each item remains consistently aligned with its most relevant interest.

Our contributions are summarized as follows:

- We empirically find that existing multi-interest sequential recommendation models often produce interest representations dominated by a few highly relevant items.
- We propose RCMIRec, a reconstructed dynamic routing method that integrates covariance routing regularization and backward-flow reconstruction to mitigate routing-weight over-concentration and better model users' diverse interests.
- We conduct extensive experiments on three public benchmark datasets to validate the effectiveness of RCMIRec.

II. Related Work

A. Sequential Recommendation

Sequential recommendation aims to predict users' future preferences based on their interaction sequences, such as browsing, clicking, and purchasing behaviors. Early studies model user preferences by treating interaction sequences as a whole, with representative methods such as collaborative filtering [12], matrix factorization [13], and Markov models [14]. However, these methods have limited capacity to capture complex and dynamic sequential dependencies.

With the advancement of deep learning, various neural architectures have been introduced into sequential recommendation. RNN-based models, such as GRU4Rec [15], effectively model temporal dependencies in user behavior sequences. Attention-based models, including SASRec [16], BERT4Rec [17], and S3Rec [18], further improve performance by capturing long-range dependencies. More recently, graph neural network-based methods [19]–[22] model user interactions as graph structures to learn more complex relational patterns. However, as pointed out by DisMIR [9], most of these methods learn a single user preference representation, which fails to capture the multiple interests commonly exhibited by users, thus limiting further performance improvements.

B. Multi-Interest Sequential Recommendation

In recent years, multi-interest learning has shown great potential in enhancing sequential recommendation performance. MIND [6] is a representative early work that introduces the dynamic routing mechanism from capsule networks to extract multiple user interest representations, enabling a more fine-grained modeling of users' diverse preferences.

Subsequent studies on multi-interest sequential recommendation can be broadly classified into two main categories. The first category leverages external auxiliary information to facilitate interest modeling. For example, UMI [7] incorporates user-side auxiliary information to generate personalized interest representations, MINER [8] utilizes item category information to align learned interests with user preferences, KEMI [23] integrates knowledge graphs to enhance semantic consistency, and MDR [24] improves preference modeling by incorporating multimodal data. While effective, these methods are heavily dependent on high-quality auxiliary data, which is often unavailable in real-world scenarios.

The second category focuses on extracting higher-order implicit information directly from user behavior sequences, without relying on external knowledge. SimEmb [25] enriches item representations by simulating attribute semantics, REMI [3] improves training through refined negative sampling strategies, and DisMIR [9] redesigns the training paradigm to disentangle multiple interests within the embedding space. Although these methods achieve notable performance gains, existing multi-interest methods still face limitations in effectively modeling diverse interests, which calls for further investigation.

III. Methodology

In this section, we first formulate the multi-interest sequential recommendation problem. Next, we introduce RCMIRec, whose overall framework is shown in Fig. 3. Finally, we provide a detailed description of each component of RCMIRec, including (1) user multi-interest learning, (2) covariance routing regularization, (3) backward-flow reconstruction, (4) multi-task training, and (5) prediction.

A. Problem Formulation

We define $U = \{u_1, u_2, \ldots, u_m\}$ and $V = \{e_1, e_2, \ldots, e_n\}$ as the sets of users and items, respectively, where m and n represent the number of users and items. For each user $u \in U$, a time-ordered interaction sequence is given by $s^{(u)} = (e_1^u, e_2^u, \ldots, e_t^u)$, where e_t^u denotes the t-th item interacted with by user u.

The goal of multi-interest sequential recommendation is to predict the next item a user may interact with by learning multiple interest representations from the user behavior sequences, ranking all candidate items accordingly, and recommending the Top-N items to each user.

B. User Multi-Interest Learning

1) Item Embedding Layer: The item embedding layer maps each item into a d-dimensional latent space to learn compact item representations for recommendation. Specifically, all item embeddings are organized into an embedding matrix as follows:

$$E_I = [e_1, e_2, \ldots, e_l] \in \mathbb{R}^{l \times d} \tag{1}$$

where $e_r \in \mathbb{R}^d$ denotes the embedding of the r-th item, and l denotes the length of the user behavior sequence.

2) Multi-Interest Extraction Layer: We employ a capsule network-based dynamic routing mechanism to extract user interest representations from the user behavior sequences. We compute the item-interest routing weights b_{ij}, which measure the similarity between the i-th item and the j-th interest capsule:

$$b_{ij} = e_i^\top \cdot S \cdot V_j \tag{2}$$

where e_i denotes the embedding of item i, V_j is the j-th interest capsule, and $S \in \mathbb{R}^{d \times d}$ is a learnable transformation matrix. All routing weights are organized into a matrix B. A softmax operation is then applied over interests to obtain routing attention weights w_{ij}:

$$w_{ij} = \frac{\exp(b_{ij})}{\sum_{j=1}^{K} \exp(b_{ij})} \tag{3}$$

where w_{ij} represents the probability that item i is assigned to interest j. Based on w_{ij}, item embeddings are aggregated to form the interest representations c_j:

$$c_j = \sum_i w_{ij} \cdot S \cdot e_i, \quad j = 1, \ldots, K \tag{4}$$

Finally, a squashing function is applied to normalize the interest capsules V_j:

$$V_j = \text{squash}(c_j) = \frac{\|c_j\|^2}{1 + \|c_j\|^2} \cdot \frac{c_j}{\|c_j\|} \tag{5}$$

The routing coefficients are initialized from a Gaussian distribution $\mathcal{N}(0, \sigma^2)$, and the dynamic routing process is iteratively performed until convergence.

3) Multi-Interest Aggregation Layer: We compute user-interest attention weights to aggregate multiple interest representations into a unified user representation. Specifically, the attention weight a_j for the j-th interest representation V_j is obtained by measuring its similarity to the average embedding $\bar{e}$ of the user behavior sequences. The final user representation v is computed as:

$$v = \sum_{j=1}^{K} a_j \cdot V_j, \quad a_j = \frac{\exp(V_j^\top \cdot \bar{e})}{\sum_{j=1}^{K} \exp(V_j^\top \cdot \bar{e})} \tag{6}$$

C. Covariance Routing Regularization

Empirical observations show that after several training epochs, the learned interest representations tend to excessively focus on individual items in the user behavior sequences, as illustrated in Fig. 2. This over-concentration causes the interest representations to rely on only a small subset of historical items, thereby weakening the expressive power of multi-interest modeling. Further analysis indicates that this issue mainly arises from the sparsity of the item-to-interest routing weight matrix. To address this issue, we introduce a covariance routing regularization term on the routing weights to suppress sparsity and encourage more balanced distributions. Variance and covariance reflect the distribution differences of items across interests, where excessively large variances indicate that certain items dominate the interest modeling process. Accordingly, we impose regularization constraints on the variance and covariance of routing weights to penalize over-concentration, guiding the model to allocate routing weights more evenly across different interests. Specifically, the regularization term is computed as follows:

$$C = (B - \bar{B})^\top \cdot (B - \bar{B}) \tag{7}$$

$$\mathcal{L}_{\text{reg}} = \|\text{diag}(C)\|_F^2 \tag{8}$$

where $\bar{B}$ is the mean of B along the first axis, $C \in \mathbb{R}^{K \times K}$ represents the covariance matrices of routing weights for different interests, diag(C) represents the extraction and construction of a diagonal matrix, and $\|\cdot\|_F$ denotes the Frobenius norm of matrices.

D. Backward-Flow Reconstruction

Although the regularization strategy encourages more balanced routing weight distributions, it does not explicitly guarantee accurate item-interest alignment, which may lead to item-interest mismatch. To address this issue, we introduce a backward-flow reconstruction mechanism to constrain the matching relationship between items and interests, ensuring that each item remains aligned with its most relevant interest during routing weight adjustment after regularization. This mechanism effectively mitigates the mismatch introduced by regularization, enabling the learned interest representations to better reflect the associated item content and capture more accurate semantic information. The computation is formulated as follows:

$$Z_j = W_1 \cdot V_j \tag{9}$$

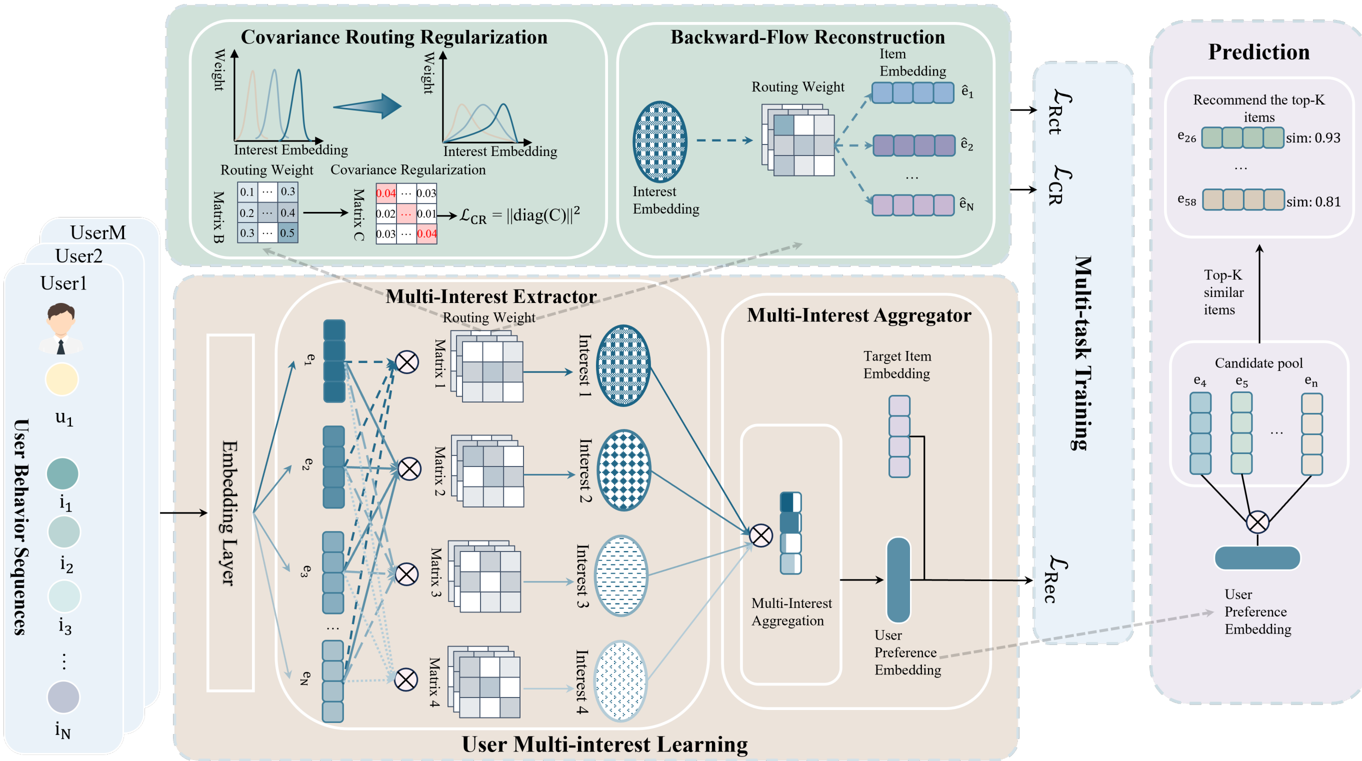

Fig. 3. The architecture of Reconstructing Dynamic Routing for Multi-Interest Sequential Recommendation (RCMIRec).

$$\beta_{j,i,k} = \frac{\exp(w_k^\top \cdot \tanh(W_2 \cdot z_{j,i}))}{\sum_m \exp(w_k^\top \cdot \tanh(W_2 \cdot z_{j,m}))} \tag{10}$$

$$\hat{e}_{j,k} = \sum_i \beta_{j,i,k} \cdot W_3 \cdot z_{j,i} \tag{11}$$

where the Upsample function is a linear projection $W_1 \in \mathbb{R}^{N\times d}$ followed by a reshape operation to transform the linearly projected vector to a matrix $Z_j \in \mathbb{R}^{N\times d}$. d is the hidden size in Re-construct backward flow. $z_{j,i}$ is the i-th unit in Z_j, $W_2 \in \mathbb{R}^{d\times d}$, and $w_k \in \mathbb{R}^d$ are learnable transformations, and each input item e_k has a corresponding w_k. We propose to reconstruct representative items corresponding to each interest. Specifically, the positive item set $\mathcal{P}_j$ is adopted as the representative items for the j-th interest. Accordingly, the loss function of the backward-flow reconstruction is formulated as follows:

$$L_{Rct} = \sum_j \sum_k (e_k \in P_j) \left\| \hat{e}_{j,k} - e_k \right\|_F^2 \tag{12}$$

E. Multi-task Training

We adopt a multi-task training strategy to jointly optimize multiple objectives, allowing different tasks to collaboratively improve model performance. The main recommendation task is optimized using the Bayesian Personalized Ranking(BPR) loss, which learns user preferences by maximizing the score difference between positive and negative items. The BPR loss is defined as:

$$\mathcal{L}_{BPR} = -\sum_{(u,i,j)\in\mathcal{D}} \log\sigma\left(v_u^\top \cdot e_{target} - \sum_{n_j=1}^{m} v_u^\top \cdot e_{n_j} \right) \tag{13}$$

where e_{target} denotes the target (positive) item, e_{n_j} represents the sampled negative items, and m is the number of negative samples.

In addition, the covariance routing regularization and the backward-flow reconstruction mechanism introduce two auxiliary objectives, namely the covariance regularization loss $\mathcal{L}_{CR}$ and the reconstruction loss $\mathcal{L}_{Rct}$. The overall training objective is formulated as:

$$\mathcal{L} = \mathcal{L}_{BPR} + \lambda_1 \cdot \mathcal{L}_{CR} + \lambda_2 \cdot \mathcal{L}_{Rct} \tag{14}$$

where λ_1 and λ_2 are balancing coefficients that control the relative contributions of different loss terms.

F. Prediction

We compute the similarity scores between the user and the items in the candidate pool are computed. By calculating the final score of each item $f(u,i)$, the Top-N items with the highest scores are recommended to the user. The computation formula is as follows:

$$f(u,i) = (v_u \cdot e_i) \tag{15}$$

Where $v_u \in \mathbb{R}^d$ denotes the user preference representation of user u, and e_i represents the embedding vector of item i in the candidate pool.

IV. EXPERIMENTS

To evaluate the effectiveness of the proposed RCMIRec method, we conduct a series of experiments on three large-scale public datasets to investigate the following research questions:

- RQ1: Does the proposed RCMIRec method outperform existing state-of-the-art multi-interest sequential recommendation models?
- RQ2: How do the individual modules in RCMIRec affect its overall performance?
- RQ3: Does the proposed RCMIRec method effectively mitigate the issue of interest representations overly relying on a small number of highly relevant items?
- RQ4: How do different hyperparameters influence the performance of the RCMIRec method?

A. Experimental Setting

1) Datasets: We evaluate the proposed RCMIRec on three large-scale public datasets: Books, Gowalla, and RetailRocket. The Books dataset is a subset of the Amazon dataset and contains user interactions related to books, including reviews, ratings, and metadata. The Gowalla dataset is collected from a location-based social networking platform and consists of user check-in records with timestamps. RetailRocket is a widely used e-commerce dataset that includes various user behaviors, such as browsing, clicking, and purchasing.

For data preprocessing, we follow the standard settings adopted in prior work [26], [27]. Users and items with fewer than five interactions are removed, and all interactions are treated as implicit feedback. The statistics of the processed datasets are summarized in Table I.

TABLE I
STATISTICS OF DATASETS

Datasets	#Users	#Items	#Interactions
Gowalla	165,506	174,605	2,061,264
Retail Rocket	33,708	81,635	356,840
Amazon Books	603,668	367,982	8,898,041

2) Baselines: We compare the proposed RCMIRec with representative baseline methods, including one single-interest and seven multi-interest sequential recommendation models. The detailed descriptions of the baseline models are as follows.

- Y-DNN [28] is a classical industrial-level sequential recommendation model that adopts a dual-tower deep neural network architecture. It aggregates users' historical interactions through pooling operations and employs a multilayer perceptron to generate user preference representations for recommendation.
- MIND [6] is one of the earliest representative methods that introduces multi-interest modeling into sequential recommendation. It utilizes a dynamic routing mechanism to extract multiple interest representations from users' historical interaction sequences and leverages capsule networks to adaptively aggregate different interests.
- ComiRec [5] proposes a multi-interest sequential recommendation framework that incorporates a multi-head attention mechanism to learn diverse user interests. In our experiments, we adopt the ComiRec-SA variant, which achieves comparable performance to ComiRec-DR while offering more stable training and higher efficiency.
- Re4 [29] presents a high-performance multi-interest recommendation framework with a three-stage optimization pipeline, which effectively enhances the diversity of user interest representations.
- UMI [7] is a multi-interest learning framework that incorporates user-side auxiliary information (e.g., demographic attributes such as gender and age). It employs an attention-gated mechanism to generate personalized multi-interest representations, thereby improving the accuracy of candidate matching.
- PIMIRec [30] explicitly models users' historical interaction sequences by jointly considering the periodicity and implicit characteristics of user–item interactions, leading to improved sequential recommendation performance.
- REMI [3] adopts an IHN-based negative sampling strategy together with routing regularization to alleviate the routing collapse problem, thereby enhancing the performance of sequential recommendation.
- DisMIR [9] is an advanced multi-interest sequential recommendation framework that redesigns the multi-interest training paradigm to effectively learn and disentangle multiple user interests, resulting in improved recommendation performance.
- PoMRec [27] introduces a prompt-based multi-interest learning framework that adapts user interaction sequences to different learning objectives, thereby enhancing sequential recommendation performance.

3) Implementation Details: All models are implemented in Python 3.12.4 using PyTorch 2.5.1. For a fair comparison, all methods adopt an embedding dimensionality of 64 and a learning rate of 1×10^{-3}, and are optimized using the Adam optimizer. The baseline models follow the optimal hyperparameter settings reported in their original papers. For RCMIRec, a hard negative sampling strategy is applied, where 10 hard negatives are sampled on the Gowalla and RetailRocket datasets, and 50 on the Books dataset. The balancing coefficients λ_1 and λ_2 are set to $[0.1,\, 0.1,\, 1.0]$ and $[0.01,\, 0.001,\, 0.01]$ on the RetailRocket, Gowalla, and Books datasets, respectively. The number of user interests K is set to $[8,\, 6,\, 6]$ for these datasets.

B. Performance Comparison(RQ1)

Table II compares the recommendation performance of RCMIRec with one single-interest model and seven multi-interest sequential recommendation models on three public datasets, with the best and second-best results highlighted in

TABLE II
MODEL COMPARISON. BOLD: BEST PERFORMANCE. UNDERLINED: SUBOPTIMAL PERFORMANCE. 'IMPROV' INDICATES THE RELATIVE IMPROVEMENTS OF RCMIREC OVER THE STRONGEST BASELINE

DATASET	Metric	Y-DNN	MIND	ComiRec	Re4	UMI	PIMIRec	REMI	DisMIR	PoMRec	RCMIRec	Improv.
Retail Rocket	R@20	0.1050	0.1171	0.1304	0.1397	0.1519	0.1828	0.2129	0.2385	0.2355	**0.2553**	7.03%
	ND@20	0.0641	0.0698	0.0689	0.0785	0.0875	0.1025	0.1198	0.1330	0.1357	**0.1417**	4.42%
	HR@20	0.1711	0.1883	0.1904	0.2103	0.2364	0.2764	0.3183	0.3524	0.3559	**0.3705**	4.11%
	R@50	0.1608	0.1899	0.1922	0.2194	0.2423	0.2811	0.3160	0.3447	0.3434	**0.3674**	6.57%
	ND@50	0.0701	0.0795	0.0786	0.0884	0.0974	0.1093	0.1281	0.1360	0.1355	**0.1408**	3.53%
	HR@50	0.2518	0.2927	0.2895	0.3174	0.3574	0.3969	0.4515	0.4815	0.4784	**0.4939**	2.58%
Gowalla	R@20	0.0864	0.0901	0.0805	0.0843	0.0961	0.1193	0.1300	0.1312	0.1352	**0.1484**	9.74%
	ND@20	0.1384	0.1331	0.1210	0.1287	0.1391	0.1603	0.1715	0.1855	0.1856	**0.2029**	9.34%
	HR@20	0.3211	0.3129	0.2901	0.3104	0.3314	0.3843	0.4021	0.4187	0.4287	**0.4626**	7.91%
	R@50	0.1388	0.1456	0.1320	0.1396	0.1642	0.1951	0.2109	0.2107	0.2157	**0.2329**	7.96%
	ND@50	0.1434	0.1424	0.1310	0.1410	0.1505	0.1660	0.1792	0.1812	0.1819	**0.2006**	10.29%
	HR@50	0.4390	0.4442	0.4086	0.4224	0.4719	0.5207	0.5498	0.5527	0.5539	**0.5997**	8.27%
Amazon Books	R@20	0.0467	0.0420	0.0557	0.0597	0.0690	0.0682	0.0826	0.0848	0.0869	**0.0915**	5.31%
	ND@20	0.0391	0.0357	0.0446	0.0476	0.0527	0.0526	0.0623	0.0648	0.0665	**0.0693**	4.16%
	HR@20	0.1043	0.0986	0.1142	0.1240	0.1423	0.1411	0.1650	0.1740	0.1767	**0.1827**	3.40%
	R@50	0.0722	0.0687	0.0863	0.0690	0.1053	0.1056	0.1189	0.1348	0.1355	**0.1400**	3.34%
	ND@50	0.0457	0.0433	0.0511	0.0576	0.0587	0.0583	0.0657	0.0742	0.0747	**0.0768**	2.80%
	HR@50	0.1607	0.1533	0.1796	0.1975	0.2059	0.2062	0.2298	0.2604	0.2613	**0.2672**	2.27%

bold and underlined, respectively. Overall, multi-interest models outperform the single-interest baseline Y-DNN, demonstrating the effectiveness of modeling users' diverse interests. However, some early multi-interest methods, such as MIND and ComiRec, perform worse than Y-DNN on certain datasets, which can be attributed to their limited interest routing capabilities and relatively immature training strategies, resulting in insufficiently discriminative interest representations. More recent multi-interest models achieve notable performance improvements by introducing more sophisticated modeling and optimization techniques. For example, Re4 enhances interest diversity through a multi-stage optimization framework, UMI incorporates auxiliary user information for personalized interest modeling, and PIMIRec captures both long-term and short-term preferences by modeling periodic and implicit user behaviors. DisMIR and PoMRec further improve performance by redesigning the training paradigm through disentangled learning and prompt-based modeling, respectively. Nevertheless, our experimental results indicate that these methods still suffer from the issue that each interest representation is often dominated by a small number of highly relevant items. In contrast, RCMIRec consistently achieves the best performance across all datasets, with average improvements of 4.71%, 8.92%, and 3.55% on RetailRocket, Gowalla, and Books, respectively. These results demonstrate the effectiveness of RCMIRec in alleviating interest dominance and highlight its superiority in multi-interest sequential recommendation.

C. Ablation Study (RQ2)

To verify the effectiveness of each component in RCMIRec, we conduct ablation studies by removing the covariance routing regularization (CR) module or the backward-flow reconstruction (BR) module, resulting in two variants: w.o.CR and w.o.BR. Table III presents a performance comparison of RCMIRec and its variants on the RetailRocket, Gowalla, and Books datasets. The results show that RCMIRec consistently outperforms both variants across all evaluation metrics, in-

TABLE III
PERFORMANCE OF VARIANT RCMIREC FOR ABLATION STUDIES

Dataset	Method	R@20	ND@20	HR@20	R@50	ND@50	HR@50
Retail Rocket	RCMIRec	**0.2553**	**0.1417**	**0.3705**	**0.3674**	**0.1408**	**0.4939**
	w.o.CR	0.2424	0.1316	0.3575	0.3442	0.1299	0.4702
	w.o.BR	0.2527	0.1350	0.3667	0.3626	0.1368	0.4889
Gowalla	RCMIRec	**0.1484**	**0.2029**	**0.4626**	**0.2329**	**0.2006**	**0.5997**
	w.o.CR	0.1298	0.1790	0.4195	0.2085	0.1817	0.5529
	w.o.BR	0.1329	0.1841	0.4286	0.2158	0.1851	0.5636
Amazon Books	RCMIRec	**0.0915**	**0.0693**	**0.1827**	**0.1400**	**0.0768**	**0.2672**
	w.o.CR	0.0845	0.0638	0.1715	0.1317	0.0724	0.2559
	w.o.BR	0.0902	0.0681	0.1806	0.1393	0.0765	0.2664

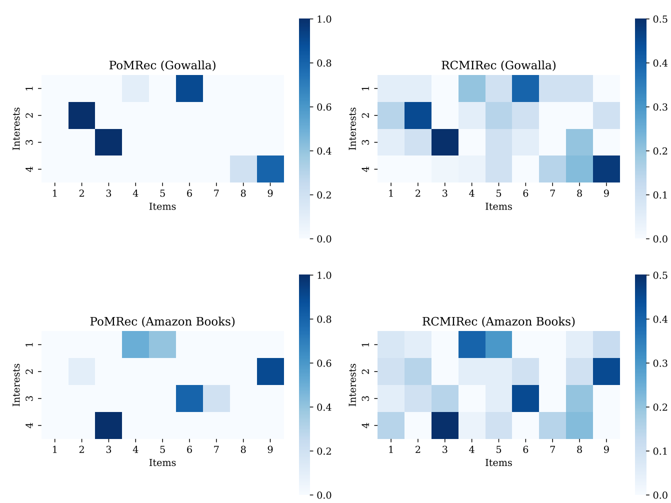

Fig. 4. Visualization of examples of item-interest routing weights for PoMRec and RCMIRec.

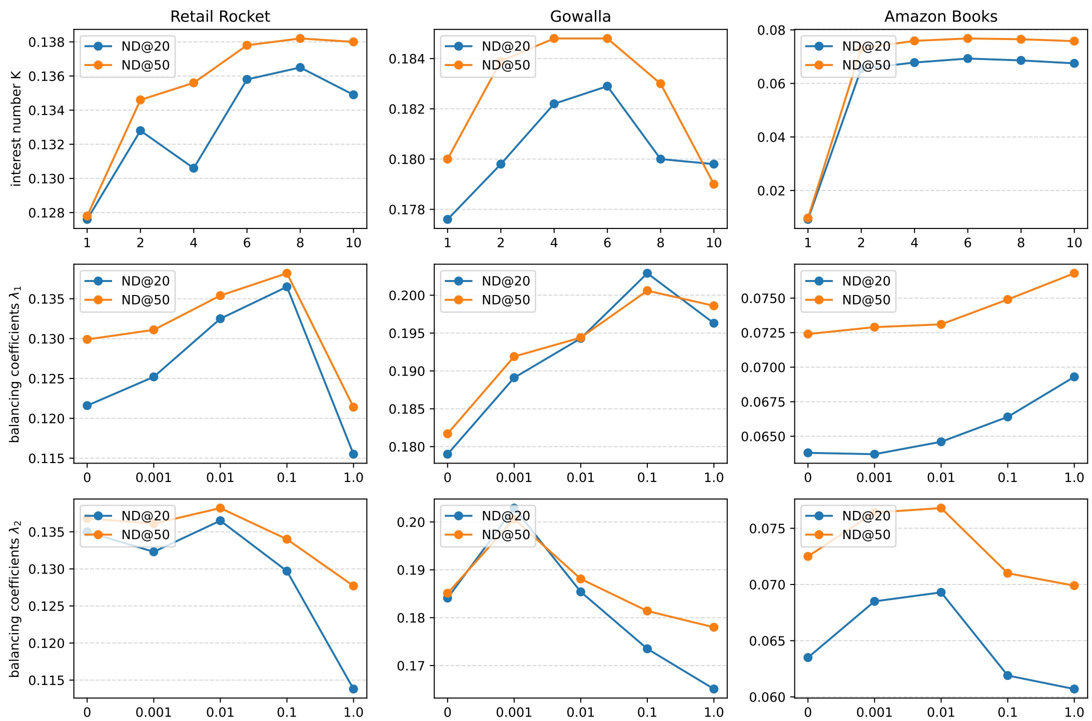

Fig. 5. Performance of RCMIRec varying with the number of interests K, and the balancing coefficients λ_1 and λ_2.

cluding Recall, NDCG, and Hit Rate, indicating that both the CR and BR modules significantly contribute to the overall performance. Specifically, the CR module promotes more balanced routing weight distributions by imposing covariance-based constraints on inter-interest routing weights, thereby alleviating excessive concentration of weight on a small number of items. Meanwhile, the BR module preserves the alignment between items and their most relevant interests during routing weight adjustment, effectively mitigating the potential item-interest mismatch introduced by regularization. Together, these components enable RCMIRec to learn more discriminative and semantically coherent interest representations, leading to superior recommendation performance.

D. Visualization Case Study(RQ3)

To provide an intuitive comparison of interest modeling behaviors, we visualize the item-interest routing weight matrices of PoMRec and RCMIRec on the Gowalla and Amazon Books datasets, as shown in Fig. 4. Each row represents an interest representation, each column corresponds to an item, and color intensity indicates the magnitude of the routing weight. As shown in the left column of Fig. 4, PoMRec learns highly sparse routing weight matrices with severe weight concentration, where each interest is dominated by only a few items, and in some cases, by a single item, especially on the Amazon Books dataset. In contrast, the right column shows that RCMIRec produces much smoother and more balanced routing weight distributions on both datasets, allocating weights across multiple relevant items for each interest. This indicates that RCMIRec effectively alleviates excessive weight concentration during dynamic routing, enabling interest representations to aggregate information from multiple related items and capture user interests more comprehensively.

E. Hyper-parameter Study(RQ4)

We conduct a sensitivity analysis on the key hyperparameters of RCMIRec to examine their impact on model performance, focusing on the number of user interests K and the balancing coefficients λ_1 and λ_2. In each experiment, only one hyperparameter is varied while all others are kept fixed.

For the number of user interests K, we fix λ_1 and λ_2 to their optimal values and vary K within the range $[1, 10]$. As shown in Fig. 5, models with $K > 1$ consistently outperform the single-interest setting across all datasets, confirming the effectiveness of multi-interest learning. The optimal value of K varies across datasets, with $K = 8$, 6, and 6 for RetailRocket, Gowalla, and Books, respectively.

To balance the different loss terms, we further analyze the impact of the balancing coefficients λ_1 and λ_2 by fixing K to its optimal value and independently varying $\lambda_1, \lambda_2 \in [0, 1]$. As shown in Fig. 5, the optimal values of λ_1 are 0.1, 0.1, and 1.0 for RetailRocket, Gowalla, and Books, respectively, while the corresponding optimal values of λ_2 are 0.01, 0.001, and 0.01.

V. Conclusion

Through extensive experiments, we demonstrate that existing multi-interest sequential recommendation models often suffer from excessive focus on a limited number of items within each interest. To address this issue, we propose

RCMIRec, which reconstructs the dynamic routing mechanism to more effectively capture users' diverse interests. Experimental results show that RCMIRec substantially alleviates interest dominance and consistently outperforms state-of-the-art multi-interest sequential recommendation methods. In this work, we focus on a single type of user behavior and do not consider heterogeneous behaviors such as clicks or favorites. In future work, we plan to explore multi-interest modeling across heterogeneous interaction sequences to achieve more comprehensive user preference modeling and further improve recommendation performance.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. 62467008,the Xinjiang Talent Development Fund through the First-Class Science and Technology Leading Talent Project of the Xinjiang Strategic Talent Training Program under Grant No. XJRC-2025-KJ-PY-KJLJ-046, and the Xinjiang University Outstanding Graduate Students Innovation Project under Grant Nos. XJDX2025YJS081 and XJDX2025YJS179.

REFERENCES

[1] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. Orgun, "Sequential recommender systems: challenges, progress and prospects," *arXiv preprint arXiv:2001.04830*, 2019.

[2] Z. Hu, S. Nakagawa, Y. Zhuang, J. Deng, S. Cai, T. Zhou, and F. Ren, "Hierarchical denoising for robust social recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[3] Y. Xie, J. Gao, P. Zhou, Q. Ye, Y. Hua, J. B. Kim, F. Wu, and S. Kim, "Rethinking multi-interest learning for candidate matching in recommender systems," in *Proceedings of the 17th ACM conference on recommender systems*, 2023, pp. 283–293.

[4] Y. Du, H. Liu, and Z. Wu, "Modeling multi-factor and multi-faceted preferences over sequential networks for next item recommendation," in *Machine Learning and Knowledge Discovery in Databases. Research Track: European Conference, ECML PKDD 2021, Bilbao, Spain, September 13–17, 2021, Proceedings, Part II 21*. Springer, 2021, pp. 516–531.

[5] Y. Cen, J. Zhang, X. Zou, C. Zhou, H. Yang, and J. Tang, "Controllable multi-interest framework for recommendation," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 2942–2951.

[6] C. Li, Z. Liu, M. Wu, Y. Xu, H. Zhao, P. Huang, G. Kang, Q. Chen, W. Li, and D. L. Lee, "Multi-interest network with dynamic routing for recommendation at tmall," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 2615–2623.

[7] Z. Chai, Z. Chen, C. Li, R. Xiao, H. Li, J. Wu, J. Chen, and H. Tang, "User-aware multi-interest learning for candidate matching in recommenders," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1326–1335.

[8] J. Li, J. Zhu, Q. Bi, G. Cai, L. Shang, Z. Dong, X. Jiang, and Q. Liu, "Miner: Multi-interest matching network for news recommendation," in *Findings of the association for computational linguistics: ACL 2022*, 2022, pp. 343–352.

[9] Y. Du, Z. Wang, Z. Sun, Y. Ma, H. Liu, and J. Zhang, "Disentangled multi-interest representation learning for sequential recommendation," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 677–688.

[10] Y. Cheng, Y. Fan, Y. Wang, and X. Li, "Accurate multi-interest modeling for sequential recommendation with attention and distillation capsule network," *Expert Systems with Applications*, vol. 243, p. 122887, 2024.

[11] Q. Li, H. Ma, W. Jin, Y. Ji, and Z. Li, "Multi-interest network with simple diffusion for multi-behavior sequential recommendation," in *Proceedings of the 2024 SIAM International Conference on Data Mining (SDM)*. SIAM, 2024, pp. 734–742.

[12] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The adaptive web: methods and strategies of web personalization*. Springer, 2007, pp. 291–324.

[13] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[14] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th international conference on World wide web*, 2010, pp. 811–820.

[15] T. Donkers, B. Loepp, and J. Ziegler, "Sequential user-based recurrent neural network recommendations," in *Proceedings of the eleventh ACM conference on recommender systems*, 2017, pp. 152–160.

[16] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *2018 IEEE international conference on data mining (ICDM)*. IEEE, 2018, pp. 197–206.

[17] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 1441–1450.

[18] K. Zhou, H. Wang, W. X. Zhao, Y. Zhu, S. Wang, F. Zhang, Z. Wang, and J.-R. Wen, "S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization," in *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020, pp. 1893–1902.

[19] J. Chang, C. Gao, Y. Zheng, Y. Hui, Y. Niu, Y. Song, D. Jin, and Y. Li, "Sequential recommendation with graph neural networks," in *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 2021, pp. 378–387.

[20] C. Ma, L. Ma, Y. Zhang, J. Sun, X. Liu, and M. Coates, "Memory augmented graph neural networks for sequential recommendation," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 5045–5052.

[21] J. Ma, J. Qin, P. Ji, Z. Yang, Y. Chen, D. Zhang, and S. Liu, "Towards consistent representations with bidirectional view alignment in graph contrastive learning for recommendation," *Neurocomputing*, p. 132092, 2025.

[22] Q. Feng, J. Qin, and J. Ma, "E-rns: Enhancing negative sample quality from gradient perspective for graph recommendation," in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2025, pp. 1–5.

[23] D. Liu, Y. Yang, M. Zhang, W. Wu, X. Xie, and G. Sun, "Knowledge enhanced multi-interest network for the generation of recommendation candidates," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 3322–3331.

[24] X. Wang, H. Chen, and W. Zhu, "Multimodal disentangled representation for recommendation," in *2021 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2021, pp. 1–6.

[25] Y. Liu, X. Zhang, M. Zou, and Z. Feng, "Attribute simulation for item embedding enhancement in multi-interest recommendation," in *Proceedings of the 17th ACM international conference on web search and data mining*, 2024, pp. 482–491.

[26] ——, "Co-occurrence embedding enhancement for long-tail problem in multi-interest recommendation," in *Proceedings of the 17th ACM Conference on Recommender Systems*, 2023, pp. 820–825.

[27] X. Dong, X. Song, T. Liu, and W. Guan, "Prompt-based multi-interest learning method for sequential recommendation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[28] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM conference on recommender systems*, 2016, pp. 191–198.

[29] S. Zhang, L. Yang, D. Yao, Y. Lu, F. Feng, Z. Zhao, T.-S. Chua, and F. Wu, "Re4: Learning to re-contrast, re-attend, re-construct for multi-interest recommendation," in *Proceedings of the ACM web conference 2022*, 2022, pp. 2216–2226.

[30] G. Chen, X. Zhang, Y. Zhao, C. Xue, and J. Xiang, "Exploring periodicity and interactivity in multi-interest framework for sequential recommendation," *arXiv preprint arXiv:2106.04415*, 2021.