

MuSeR: Scalable Long-sequence Recommendation with Multi-interest Modeling

Yongkang Fu¹, Beining Bao², Yu Jiang³, Xiangyu Zhao², Hongyang Wei¹, Guangxing Chen¹, Zuodong Yang¹, Shantao Li¹, Zonggang Wu¹, Yuqi Lu¹, Shouke Qin¹, Hanmeng Liu¹, Maolin Wang^{2,*}

¹Baidu, Beijing, China

²City University of Hong Kong, Hong Kong SAR, China

³Chinese University of Hong Kong, Hong Kong SAR, China

Abstract—Ultra-long user behavior sequences carry rich signals of stable and diverse preferences, yet industrial recommender systems typically truncate histories to a few hundred actions under strict latency and memory budgets, leaving long-term interests under-utilized. Users also pursue multiple heterogeneous intents across modalities such as news, Q&A, and short video, which sparse ID embeddings alone struggle to represent. We present Multi-interest Sequence Representation (MuSeR), a retrieval framework built on the deployed MGS system, which integrates three components: (i) *hierarchical temporal compression*, which retains recent actions at full resolution while progressively pooling older segments, so that per-user histories of 10^4 – 10^5 interactions fit within a fixed serving budget; (ii) *disentangled multi-query interest extraction* with orthogonality regularization; and (iii) *multimodal semantic alignment*, which augments sparse item IDs with textual summaries distilled from a large language model. For industrial deployment, MuSeR further adopts asynchronous user-representation refresh with adaptive caching and hierarchical beam-search retrieval across heterogeneous hardware. On three public benchmarks and a large-scale industrial dataset, MuSeR consistently improves Recall@K over strong long-sequence and multi-interest baselines. In online A/B tests on Baidu APP’s homepage feed, discovery feed, and short-video scenarios, MuSeR yields +0.26% daily active users and +0.89% total session duration (both statistically significant, $p < 0.05$), alongside reduced serving latency and cost. Rather than proposing a new modeling primitive, our contribution is a system-level integration that makes long-term, multi-interest, and multimodal modeling jointly deployable in a real-time production pipeline, together with the engineering practices required to sustain it.

Index Terms—User Interest Modeling, Ultra-long Behavioral Sequences, Recommender Systems, Multi-interest Modeling, Industrial Recommendation

I. INTRODUCTION

In recent years, large-scale personalized recommendations have become a cornerstone of mobile super-app ecosystems. In the Baidu APP, the first recommendation homepage, discovery feed, and short-video scenes collectively serve hundreds of millions of daily active users, accounting for the majority of total session duration. Delivering high-quality recommendations in such complex environments requires the system to process user intents with millisecond-level responsiveness while sustaining ultra-high throughput across heterogeneous hardware infrastructures [1]. At the core of personalization is the accurate modeling of user behavioral sequences. Over

years of interactions, users accumulate up to 10^5 actions across modalities such as news, QA, and short-video streams, forming ultra-long behavioral logs that encode rich, nuanced, and stable preferences. However, due to online latency constraints and resource limitations, current production systems typically truncate these to only a few hundred recent actions, leading to under-utilization of long-term interests and a reduced ability to capture stable engagement signals and diversified preferences [2], [3].

Despite progress in deep sequence modeling and retrieval [3], several unresolved challenges still hinder industrial-scale deployment of ultra-long sequence modeling. First, computational scalability and efficiency remain major bottlenecks: transformer-based models scale quadratically with input length, making histories of 10^5 actions per user infeasible under typical production SLAs (usually < 300 ms) [2]. Second, interest diversity and temporal drift amplify modeling complexity: users often exhibit multiple, sometimes conflicting interests across heterogeneous modalities, where long-term preferences evolve slowly but short-term intentions shift rapidly, requiring a unified mechanism to capture both stability and novelty. Third, an alignment gap between sparse IDs and multimodal content limits representation quality: most approaches rely on sparse item embeddings and underutilize semantic signals from text or video, so sequence compression can lose critical semantics, particularly in cold-start and short-video scenarios. Finally, engineering and deployment constraints further complicate adoption: practical recommender systems must balance accuracy gains with asynchronous computation, caching, and load-adaptive serving to ensure cost-efficient, reliable operation across heterogeneous hardware.

Recognizing these challenges, we present **Multi-interest Sequence Representation (MuSeR)**, an extension of the MGS (Multi-target Graph-based Search) retrieval framework that unifies **ultra-long sequence compression**, **multi-query disentangled interest modeling**, and **multimodal semantic alignment**. MuSeR advances industrial recommendation along three principal dimensions.

First, we design a **hierarchical temporal compression mechanism** that condenses ultra-long user histories into compact yet informative sequences. Recent actions are preserved with full granularity, mid-term behaviors are downsampled with moderate pooling, while early histories are aggressively

* Corresponding author: Maolin Wang (morin.wang@my.cityu.edu.hk).



(a) Homepage "Recommended" (b) Discovery and short-video

Fig. 1: Deployment scenarios of MuSeR in the Baidu APP and its position in the recommendation pipeline. (a) The homepage "Recommended" feed, where MuSeR serves as the retrieval module to recall high-quality candidates from ultra-long behavior logs. (b) The Discovery feed and short-video entrance, where MuSeR supports multi-interest matching under heterogeneous modalities and shifting intents.

compressed into coarser representations. This strategy ensures that the system remains scalable while still covering both short-term and long-term behavioral patterns [2], [4]. Second, we introduce a **multi-query interest extractor**, inspired by query-based representation learning frameworks. Instead of producing a single aggregated embedding, MuSeR learns multiple query vectors that distill different aspects of a user's preferences into several distinct embeddings [5]–[7]. During training, each embedding is responsible for predicting diverse future actions; in serving, the framework dynamically selects the most relevant interest vector to interact with retrieval candidates. This mechanism allows the system to capture heterogeneous and even conflicting user intents across homepage browsing, discovery feeds, and short-video scenarios [8], [9]. Third, MuSeR integrates **multimodal alignment** to bridge the gap between sparse item identifiers and rich content semantics. Beyond ID-level embeddings, we incorporate textual summaries generated by ERNIE-4.0-Turbo (distilled into its lightweight, cost-efficient variant) and semantic representations learned through BGE embeddings. The result is a unified multimodal representation that enhances recall quality, strengthens cold-start handling, and improves recommendations in multi-domain environments.

From a system engineering perspective, MuSeR couples model innovation with a cost-aware deployment pipeline. Through an *asynchronous computation strategy*, ultra-long sequence representations are periodically refreshed offline or via cache, with refresh frequencies dynamically adjusted by QPS and service load, while online scoring only processes short sequences. In retrieval, we adopt a *hierarchical beam-search strategy* adapted from HNSW, combining efficient vector similarity with deep multi-target scoring. This hybrid

design ensures high recall without compromising latency, and is well-suited for heterogeneous CPU/GPU clusters in real production environments.

We validated MuSeR through extensive A/B testing in Baidu APP's homepage, discovery feed, and short-video scenarios (as shown in Figure 1). **Since August 2025, MuSeR has been fully deployed in production**, demonstrating consistent online improvements with feed DAU lifted by +0.26% and total session duration increased by +0.89%. Beyond numerical gains, MuSeR effectively leveraged per-user histories of up to 10^5 interactions to surface deeper, more stable interests, and successfully applied multimodal alignment to improve semantic recall, confirming its scalability and robustness at industry scale. Specifically, we make the following contributions:

- We propose **MuSeR**, a novel end-to-end long-sequence recommendation framework that unifies hierarchical compression, multi-query interest extraction, and multimodal semantic alignment [10].
- We design a hierarchical temporal compression mechanism and a multi-query extractor that jointly capture both long-term stable interests and short-term evolving preferences, improving user modeling depth [11], [12].
- We develop a robust multimodal alignment pipeline based on ERNIE-4.0-Turbo and BGE, mapping sparse item IDs into dense semantic vectors that enhance retrieval quality, particularly in cold-start and short-video scenarios.
- We implement an asynchronous computation pipeline and hierarchical beam-search retrieval, enabling efficient large-scale deployment across heterogeneous hardware.

Significance. Long user behavior sequences contain rich signals of interest evolution and dependency, yet industrial recommender systems often fail to fully leverage them due to computational and latency constraints [3]. **MuSeR** unifies *long-sequence compression*, *multi-interest modeling*, and *multimodal alignment* within a deployable retrieval framework, effectively mitigating issues like interest forgetting and limited representation capacity. By balancing retrieval efficiency and personalization accuracy, MuSeR marks a key advancement in **production-ready recommender systems with stronger adaptability and expressive power**.

II. METHOD

We introduce the technical details of the proposed **MuSeR** framework in this section. We begin with an overview of the overall architecture, followed by detailed explanations of its key components, including hierarchical temporal compression, multi-query interest extraction, and multimodal semantic alignment. Finally, we describe the inference and optimization.

A. Overview

This section briefly introduces the overall pipeline of **MuSeR**, with the framework overview illustrated in Figure 2. Building upon the industrial MGS framework mentioned in [13], where the implementation details were not elaborated, MGS can be regarded as a NANN-style retrieval system based on HNSW, supporting multi-objective indexing and

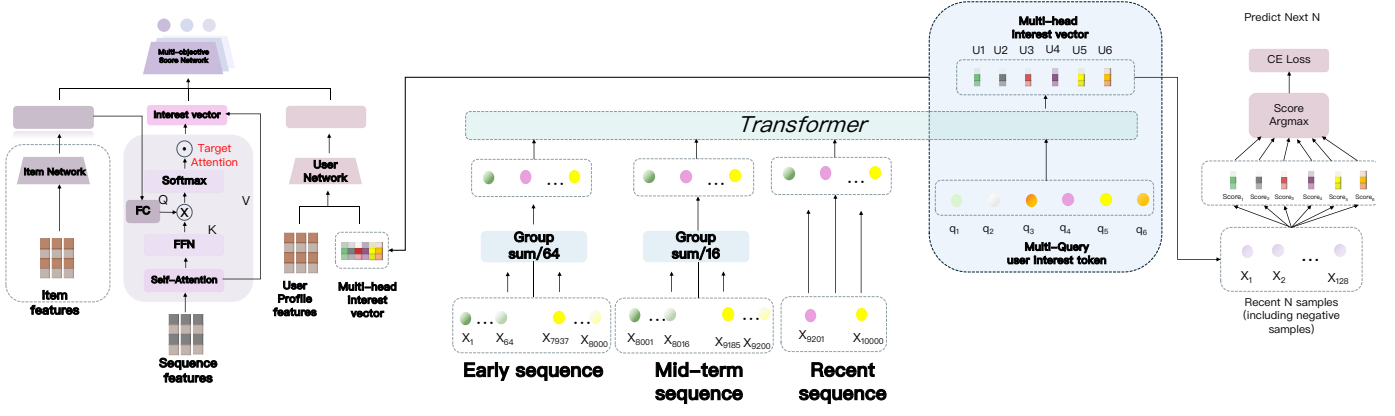


Fig. 2: Overview of the proposed **MuSeR** framework, which integrates hierarchical temporal compression, multi-query interest extraction, and multimodal semantic alignment for efficient user modeling and retrieval.

search. Based on this retrieval architecture, MuSeR further unifies ultra-long user behavior modeling, multi-interest representation learning, and multimodal alignment within a single framework. The system follows a hybrid offline–online computation strategy: offline modules asynchronously encode users’ long-term behavioral sequences into multiple interest embeddings, while the online module continuously updates short-term session behaviors for real-time retrieval.

Formally, given a user u with an interaction sequence $S_u = [i_1, i_2, \dots, i_T]$ where $i_t \in \mathcal{I}$ denotes an interacted item, the learning objective is to estimate the matching function:

$$\hat{y}_{ui} = F_{\Theta}(S_u, i), \quad (1)$$

where F_{Θ} maps an ultra-long sequence into a user representation u and predicts the interaction probability between u and item i . To achieve both accuracy and efficiency, MuSeR contains three major modeling phases described below.

B. Hierarchical Temporal Compression

Ultra-long user histories (typically 10^4 – 10^5 items) contain rich behavioral patterns but are expensive to process with Transformer-based models [14]. We thus introduce a hierarchical temporal compression mechanism to preserve multi-scale temporal information while maintaining computational tractability.

a) *Temporal segmentation.*: The full sequence S_u is partitioned into three chronological segments:

$$S_u = [S_u^{(r)}, S_u^{(m)}, S_u^{(e)}], \quad (2)$$

corresponding to recent ($S_u^{(r)}$), mid-term ($S_u^{(m)}$), and early ($S_u^{(e)}$) behaviors. The recent segment remains intact to maintain high-resolution dynamics, while earlier segments are gradually pooled with increasing stride to suppress redundancy.

b) *Progressive pooling.*: We apply different down-sampling rates for different temporal regions:

$$S_u^{(r)} = [i_1, \dots, i_{N_r}], \quad (3)$$

$$S_u^{(m)} = \text{Pool}_{16}(i_{N_r+1:N_m}), \quad (4)$$

$$S_u^{(e)} = \text{Pool}_{64}(i_{N_m+1:T}), \quad (5)$$

where $\text{Pool}_k(\cdot)$ denotes sum or attentive pooling over k neighboring items. After compression, the total effective sequence length can be represented as:

$$L_c = N_r + \frac{N_m - N_r}{16} + \frac{T - N_m}{64}. \quad (6)$$

c) *Sequence encoding.*: Each token is represented as the sum of item and positional embeddings:

$$\mathbf{x}_t = \mathbf{E}_{\text{id}}(i_t) + \mathbf{E}_{\text{pos}}(t), \quad (7)$$

then encoded by a lightweight Transformer encoder:

$$\mathbf{H}_u = \text{TransformerEnc}([\mathbf{x}_1, \dots, \mathbf{x}_{L_c}]). \quad (8)$$

The output $\mathbf{H}_u \in \mathbb{R}^{L_c \times d}$ effectively captures both short-range and long-range dependencies, preserving long-term interest signals at low computational cost.

C. Multi-Query Interest Extraction

User intent is often inherently multi-faceted: for example, the same user may simultaneously consume news, shop, or browse short videos. To uncover this heterogeneity, we introduce a multi-query attention module that decodes multiple interest embeddings from $\mathbf{H}_u$.

A set of M learnable query vectors $\{q_1, \dots, q_M\}$ attend to the contextual states through scaled dot-product attention:

$$\mathbf{u}_m = \text{softmax}\left(\frac{q_m \mathbf{H}_u^T}{\sqrt{d}}\right) \mathbf{H}_u, \quad m = 1, \dots, M. \quad (9)$$

Each $\mathbf{u}_m$ encodes a distinct behavioral aspect derived from different temporal or contextual patterns. During the offline training phase, all $\{\mathbf{u}_m\}$ jointly predict future user behaviors; during online inference, the retrieval system adaptively selects the most relevant interest vector according to the given candidate item embedding e_i :

$$\alpha_m = \frac{\exp(\text{sim}(\mathbf{u}_m, e_i)/\tau)}{\sum_n \exp(\text{sim}(\mathbf{u}_n, e_i)/\tau)}, \quad \mathbf{u} = \sum_m \alpha_m \mathbf{u}_m. \quad (10)$$

Here, $\text{sim}(\cdot)$ denotes cosine similarity and τ is a temperature parameter controlling interest sharpness.

This mechanism allows MuSeR to align fine-grained interests with corresponding candidate categories dynamically.

D. Multimodal Semantic Alignment

Purely ID-based representations lack semantic generalization and perform poorly for cold-start or sparse items. To mitigate this, we enhance items with multimodal semantic embeddings that align text- and content-level information with user representations.

Each item's textual description is first processed by a powerful large language model (ERNIE-4.0-Turbo) through chain-of-thought prompting, generating condensed and informative semantic perspectives. We then distill this process into a lightweight variant (ERNIE-Speed) to scale cost-efficiently, and project the resulting summaries into dense semantic vectors using the BGE embedding model. Finally, we fuse the content and ID embeddings as follows: $e_i = W_1 e_i^{(id)} + W_2 e_i^{(sem)}$. This multimodal projection effectively maps user intent vectors and item semantics into a unified shared embedding space, facilitating semantically meaningful and accurate recalls across diverse recommendation scenarios.

E. Training Objective

MuSeR optimizes a next- K behavior prediction objective, leveraging multiple interest embeddings to forecast users' immediate future interactions. Given positive items $\mathcal{P}_u$ and sampled negatives $\mathcal{N}_u$, the training loss combines contrastive retrieval and orthogonality regularization as follows:

$$\mathcal{L}_{pred} = - \sum_{m=1}^M \sum_{i^+ \in \mathcal{P}_u} \log \frac{\exp(\text{sim}(\mathbf{u}_m, \mathbf{e}_{i^+})/\tau)}{\exp(\text{sim}(\mathbf{u}_m, \mathbf{e}_{i^+})/\tau) + \sum_{i^- \in \mathcal{N}_u} \exp(\text{sim}(\mathbf{u}_m, \mathbf{e}_{i^-})/\tau)} \quad (11)$$

$$\mathcal{L}_{orth} = \frac{1}{M^2} \sum_{m \neq n} \|\mathbf{u}_m^\top \mathbf{u}_n\|_2^2 \quad (12)$$

$$\mathcal{L} = \mathcal{L}_{pred} + \lambda \mathcal{L}_{orth} \quad (13)$$

The first term maximizes mutual similarity between interest representations and ground-truth positives, while the second encourages decorrelation across different interest heads to ensure diversity. Training uses Adam with warm-up, dropout, and mixed precision to maintain convergence and efficiency.

III. ONLINE DEPLOYMENT

This section outlines the online deployment framework of MuSeR, highlighting the integration of **asynchronous computation** for ultra-long sequence modeling and **multi-objective retrieval** optimization in large-scale industrial environments. As illustrated in Fig. 3, the proposed architecture combines offline index construction, asynchronous user representation updates, and multi-objective scoring to achieve efficient and scalable online retrieval.

A. Asynchronous Computation for Ultra-long Sequences

In the industrial environment, user behavior sequences often span tens of thousands of interactions, making real-time modeling computationally prohibitive. To address this challenge, we deploy an **asynchronous cache-based computation framework**. The ultra-long sequence encoder and multi-head interest extraction module operate under a dual-frequency

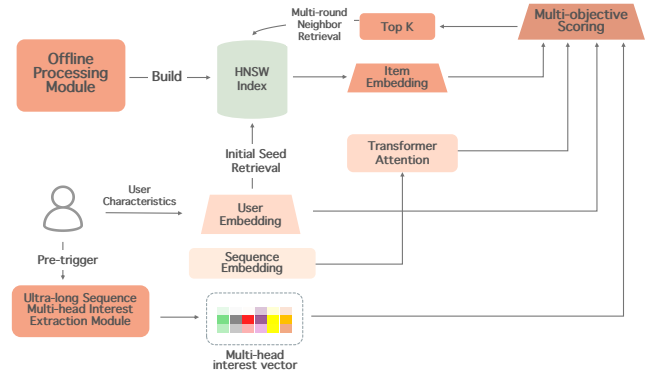


Fig. 3: Overview of the online deployment architecture. The framework integrates asynchronous ultra-long sequence modeling with multi-objective retrieval. The offline processing module constructs the HNSW index, while real-time user embeddings and multi-head interest extraction interact with transformer attention and a multi-objective scoring network for efficient large-scale retrieval.

mechanism: high-frequency updates for recent short-term sequences and low-frequency asynchronous refresh for long-term interest embeddings.

Specifically, user representations $\{\mathbf{u}_1^{(long)}, \dots, \mathbf{u}_M^{(long)}\}$ are computed offline or refreshed periodically according to traffic load. These embeddings are stored in distributed caches and reused during online inference. Real-time requests only trigger the rapid computation of short-term sequence encoder outputs $\mathbf{u}^{(short)}$, which are dynamically fused with cached embeddings via a weighted aggregator as follows:

$$\mathbf{u}^{(final)} = \beta(t) \mathbf{u}^{(short)} + [1 - \beta(t)] \mathbf{u}^{(long)}, \quad (14)$$

where the fusion weight $\beta(t)$ is determined by adaptive load monitoring and service concurrency. The system dynamically throttles the QPS (queries per second) of asynchronous updates to maintain an optimal trade-off between model effectiveness and computational cost. This heterogeneous-frequency computation pipeline reduces redundant encoding by up to 60%, ensuring both low latency and stable retrieval quality under real-world traffic fluctuations.

B. Multi-objective Retrieval and Efficient Vector Indexing

The online retrieval service extends the standard MGS framework with a multi-objective search paradigm inspired by HNSW [15]. We construct a multi-layer hierarchical vector index graph using Euclidean distance as the primary similarity criterion for candidate linkage and neighbor selection. During retrieval, a complex DNN-based scoring network jointly evaluates multiple objectives such as semantic relevance, user engagement probability, and commercial conversion potential, enabling more precise and balanced candidate ranking.

A key challenge arises when the index-building metric (Euclidean distance) differs from the ranking metric of the multi-objective model, causing recall inefficiency. To mitigate this misalignment, we redesign the retrieval process as a

hierarchical beam-search strategy. At each layer, candidate nodes are expanded by top- k anchors and re-scored with fine-grained objectives. The beam width and expansion ratio are adaptively tuned through parameter exploration, aligning graph topology with model scoring. This improvement significantly reduces recall loss while maintaining high throughput. Furthermore, the proposed retrieval mechanism efficiently exploits heterogeneous hardware environments. The graph index construction runs on high-capacity CPU clusters, while real-time vector traversal and scoring are executed on GPU or Kunlun NPU with mixed-precision acceleration. The new hierarchical beam-search approach automatically balances computation across tiers, achieving hardware-level parallelism with minimal memory overhead. Empirically, this strategy effectively reduces end-to-end retrieval latency by 27% and lowers overall engineering deployment cost by over 35% compared with conventional flat HNSW indexing.

C. Representation and Cross-model Fusion

Our retrieval architecture combines the **dual-tower model** and **deep cross-network** advantages into one unified retrieval system. The document tower fully utilizes side information (e.g., category, semantic, and multimodal signals), producing compact embeddings for high-throughput retrieval. Meanwhile, the user tower, equipped with multi-head interest extraction, provides diverse high-resolution behavioral vectors. The DNN interaction layer within the multi-objective scoring model effectively captures cross-feature dependencies such as historical-semantic alignments. This hybrid design enhances modeling precision and extends the recall boundary, substantially increasing relevance coverage and improving retrieval gains in online production.

Overall, the deployed online system realizes an **asynchronous, hierarchical, and multi-objective retrieval framework** that maintains millisecond-level latency while sustaining billion-scale indexing. The combination of cache-based asynchronous updates and adaptive beam-search retrieval enables flexible trade-offs between recall quality and system cost, ensuring robust and efficient recommendation performance in large-scale industrial settings.

D. Production Deployment Details

In early 2025, the MuSeR framework was rolled out to the production environment of the Baidu APP, first serving the homepage recommendation stream and then being extended to the discovery feed and the short-video feed. This rollout was validated through large-scale online A/B testing across these core traffic scenarios. Experiments were conducted on live traffic with user-level bucketing, where buckets are assigned by a hash of the user ID and held fixed throughout the experiment window; control and treatment differ only in the retrieval channel under study, while all downstream ranking and mixing stages remain identical. Each experiment ran continuously for multiple weeks over tens of millions of daily users per bucket, and the reported lifts are statistically significant under a two-sided test. In the same experiment on the homepage

feed, MuSeR increased feed Daily Active Users (DAU) by **+0.26%** and total session duration by **+0.89%**, indicating that the framework improves not only immediate interaction outcomes but also sustained user engagement. At this traffic scale, and given that the feed is already served by a mature ensemble of retrieval sources, a gain of this magnitude from a single retrieval channel is a meaningful improvement. These results confirm that MuSeR’s unified architecture, combining **hierarchical temporal compression** over per-user histories of up to 10^5 interactions with **multi-modal semantic alignment**, can translate modeling advances into measurable business impact under strict industrial constraints (e.g., latency budgets, heterogeneous hardware, and high-QPS traffic).

Concretely, MuSeR improves retrieval quality by jointly addressing two long-standing limitations in production recommenders. First, it mitigates long-history underutilization by compressing ultra-long sequences in a time-aware, information-preserving manner, enabling the system to exploit stable long-term preferences without exceeding online compute budgets. In production, long-term user representations are computed off the serving path and cached, so the ultra-long history never enters the request-time compute budget. Second, it reduces the semantic gap inherent in sparse ID-based representations through multi-modal alignment, allowing the retriever to better capture content similarity and user intent at a semantic level, which is particularly useful for recently published items, rich-media content, and cross-domain consumption patterns. Together, these capabilities enable MuSeR to provide higher-recall candidate sets with stronger relevance and robustness, which in turn contributes to the observed improvements in engagement metrics.

The specific application scenarios are illustrated below. Figure 1a shows the homepage “Recommended” feed, which presents a mixture of long-form news articles and rich-media cards. In this context, MuSeR leverages long-term user histories to infer enduring interests (e.g., recurring topics, preferred sources, and stable content styles) while still remaining responsive to recent interactions. By combining cached long-term representations with fine-grained short-term signals, MuSeR delivers personalized recommendations that retain topical diversity, supporting broader exploration within a user’s interest space. Figure 1b depicts the discovery feed and short-video grid, a highly dynamic environment characterized by rapid intent shifts, short feedback cycles, and strong sensitivity to freshness and context. Here, MuSeR’s ability to fuse real-time session signals with cached long-term interests is crucial for maintaining both instant relevance and recall stability as user goals change quickly (e.g., from entertainment to information seeking). Moreover, the use of multi-modal representations strengthens matching between users and rich video candidates beyond sparse IDs, improving semantic recall under heterogeneous modalities; this effect is more pronounced in the short-video setting, where item lifecycles are short and per-item interaction histories are correspondingly thin.

TABLE I: Performance comparison of different models on three public datasets, including Instrument, Scientific, and Game. The best and second-best results are highlighted in bold and underlined font, respectively. “*” denotes statistically significant improvements (i.e., two-sided t -test with $p < 0.05$) over the best baseline.

Methods	Instrument				Scientific				Game			
	Recall@5	Recall@10	NDCG@5	NDCG@10	Recall@5	Recall@10	NDCG@5	NDCG@10	Recall@5	Recall@10	NDCG@5	NDCG@10
Caser	0.0242	0.0392	0.0154	0.0202	0.0172	0.0281	0.0107	0.0142	0.0346	0.0567	0.0221	0.0291
GRU4Rec	0.0345	0.0537	0.0220	0.0281	0.0221	0.0353	0.0144	0.0186	0.0522	0.0831	0.0337	0.0436
HGN	0.0319	0.0515	0.0202	0.0265	0.0220	0.0356	0.0138	0.0182	0.0423	0.0694	0.0266	0.0353
SASRec	0.0341	0.0530	0.0217	0.0277	0.0256	0.0406	0.0147	0.0195	0.0517	0.0821	0.0329	0.0426
BERT4Rec	0.0305	0.0483	0.0196	0.0253	0.0180	0.0300	0.0113	0.0151	0.0453	0.0716	0.0294	0.0378
FMLP4-Rec	0.0328	0.0529	0.0206	0.0271	0.0248	0.0388	0.0158	0.0203	0.0535	0.0860	0.0331	0.0435
FDSA	0.0364	0.0557	0.0233	0.0295	0.0261	0.0391	0.0174	0.0216	0.0485	0.0857	0.0353	0.0453
S ³ -Rec	0.0340	0.0538	0.0218	0.0282	0.0253	0.0410	0.0172	0.0218	0.0533	0.0823	0.0351	0.0444
SID	0.0319	0.0438	0.0237	0.0275	0.0155	0.0234	0.0103	0.0129	0.0480	0.0693	0.0333	0.0401
CID	0.0352	0.0507	0.0234	0.0285	0.0192	0.0300	0.0123	0.0155	0.0497	0.0748	0.0343	0.0424
TIGER	0.0368	0.0574	0.0242	0.0308	0.0275	0.0431	0.0181	0.0231	0.0570	0.0895	0.0370	0.0471
TIGER-SAS	0.0375	0.0576	0.0242	0.0306	0.0272	0.0435	0.0174	0.0227	0.0561	0.0891	0.0363	0.0469
LETTER	0.0372	0.0581	0.0243	0.0310	0.0276	0.0433	0.0179	0.0230	0.0576	0.0901	0.0373	0.0475
NANN	<u>0.0373</u>	<u>0.0584</u>	<u>0.0250</u>	<u>0.0313</u>	<u>0.0281</u>	<u>0.0435</u>	<u>0.0185</u>	<u>0.0235</u>	<u>0.0582</u>	<u>0.0910</u>	<u>0.0381</u>	<u>0.0481</u>
MuSeR	0.0381*	0.0587	0.0258*	0.0319*	0.0288*	0.0441*	0.0188*	0.0238	0.0592	0.0919*	0.0389*	0.0488*

IV. EXPERIMENTS

We present experimental settings and extensive empirical results of **MuSeR** in this section.

A. Experiment Settings

Dataset. We evaluate on three subsets of the Amazon 2023 review data (Instruments, Video Games, Industrial & Scientific) [16], [17]. We apply 5-core filtering [18], build temporally ordered sequences truncated/padded to length 50, and split the data chronologically into 70%/10%/20% for train/validation/test. We also report results on a large-scale proprietary dataset from Baidu’s information feed, containing one month of real-world user–item interaction logs with about 100M users and 10M items; we train on the first 29 days and test on the last day.

Model. Our model is a 6-layer Transformer encoder (hidden size $d=512$, 8 attention heads, dropout 0.1) [14] with 6-query interest extraction (temperature $\tau=0.07$), sinusoidal positional encodings (max length 10k) [14], and multimodal fusion over ID and content features. For text/content semantics, we distill ERNIE-4.0-Turbo summaries and use BGE-base-en embeddings (768d) [19] as well as Visualized-BGE/VISTA for hybrid multimodal alignment when applicable [20]. Fusion weights are initialized to 0.7/0.3 and learned end-to-end. We include an orthogonality regularization on multi-interest heads with coefficient $\lambda=0.01$ (a common technique to decorrelate factors in multi-head/mixture settings, also adopted in multi-interest retrieval literature such as [21]).

Optimization and training. We use AdamW [22] with initial learning rate $1e-4$, weight decay $1e-5$, cosine-annealing schedule with 1k warm-up steps [23], gradient clipping at 1.0 [24], and batch sizes of 256 (public datasets) and 4096 (industrial datasets), employing mixed precision when available. Public baselines are tuned under a unified protocol.

Baselines. We compare with Caser [25], GRU4Rec [26], SASRec [27], BERT4Rec [28], FMLP-Rec [29], S³-Rec [30], TIGER [31], and NANN [32] using identical preprocessing and splits. Each method is run with 5 random seeds; we conduct two-sided t -tests with $p<0.05$ for statistical significance.

Infrastructure. Our implementation uses PyTorch 1.13.1 [33], HuggingFace Transformers [34] for text components, and Horovod for distributed training [35] with NCCL backend and CUDA/cuDNN. Mixed-precision training uses NVIDIA Apex when applicable. Inference for the long-sequence encoder runs on NVIDIA A10 (18 cards) accelerated by TensorRT FP16. Training is conducted on NVIDIA A100 (4 cards) and Baidu Kunlun P800 NPUs; large-scale training scales to H20×8 or Kunlun P800×16. Retrieval employs Kunlun R200 (293+151) with PaddlePaddle/PaddleInference for serving. The asynchronous computation pipeline uses Redis for distributed caching and Apache Kafka for streaming, with Consul for service discovery.

Metrics. Beyond standard Recall@K and NDCG@K, we also report MRR, HR, and Coverage to assess ranking quality and catalog utilization. For industrial scenarios, we further track online metrics including CTR, CVR, and engagement signals such as dwell time and session duration.

We also report results on a large Baidu APP feed dataset comprising ~ 100 M active users and 10M items, yielding ~ 5 B daily interactions. We use days 1–29 for training and day 30 for testing, and evaluate in two realistic scenarios: Home Feed (personalized long-term engagement) and Discovery/Short-Video (fast-shifting short-term intents). Online metrics include CTR/CVR and engagement (dwell time, session duration), in addition to offline retrieval metrics.

B. Performance Analysis on Public Datasets

As shown in Table I, the experimental results on three public benchmark datasets demonstrate the superior performance of **MuSeR** across diverse recommendation scenarios. On the **Instrument** dataset, **MuSeR** achieves the best performance across all metrics, with Recall@5 of 0.0381, Recall@10 of 0.0587, NDCG@5 of 0.0258, and NDCG@10 of 0.0319, showing statistically significant improvements over the strongest baseline NANN in most metrics ($p < 0.05$). Similarly, on the **Scientific** dataset, **MuSeR** consistently outperforms all competing methods, achieving Recall@5 of 0.0288, Recall@10 of 0.0441, NDCG@5 of 0.0188, and NDCG@10

of 0.0238, with statistically significant gains over NANN in Recall@5, Recall@10, and NDCG@5. For the **Game** dataset, MuSeR again establishes new state-of-the-art results with Recall@10 of 0.0919, NDCG@5 of 0.0389, and NDCG@10 of 0.0488, demonstrating significant improvements over the second-best performer, NANN.

The consistent improvements across all three datasets validate several key aspects of MuSeR’s design. First, the hierarchical temporal compression mechanism effectively captures both short-term and long-term user preferences without losing critical behavioral signals. Second, the multi-query interest extraction successfully disentangles diverse user intents, enabling more accurate representation of complex user behavior patterns. Third, the multimodal semantic alignment enhances the model’s ability to understand item semantics beyond sparse ID embeddings, particularly beneficial for cold-start scenarios and cross-domain recommendations. Notably, MuSeR’s performance gains are most pronounced on metrics like Recall@5 and NDCG@5, indicating that the framework excels at identifying the most relevant items for users, a crucial capability for practical recommendation systems where top-ranked results have the highest impact on user satisfaction. The statistical significance of these consistent improvements across multiple benchmark datasets confirms that MuSeR’s unified approach to long-sequence modeling, multi-interest representation, and multimodal alignment provides robust and broadly generalizable benefits for sequential recommendation.

C. Overall Performance on the Private Dataset

On Baidu’s large-scale private industrial dataset, **MuSeR** consistently achieves the strongest recall across both the Home Feed and the Discovery and Short Video scenarios. Compared with established baselines including SASRec, NANN, TIGER, and KuaiFormer, MuSeR delivers higher Recall@100 and Recall@500 in both settings, as reported in Table II and Table III. These results substantiate that unifying hierarchical temporal compression, multi-query interest extraction, and multimodal semantic alignment can simultaneously preserve long-horizon behavioral coverage and high-resolution short-term intent modeling. This unified design expands the effective recall boundary under industrial latency and cost constraints by reducing redundancy in ultra-long sequences while retaining diverse intent signals. In the Discovery and Short Video scenario, where user preferences evolve rapidly and items span heterogeneous modalities, the gains are pronounced, indicating that MuSeR better disentangles conflicting interests and aligns sparse identifiers with semantic content.

Overall, the consistent improvements in Table II and Table III affirm that scalable long-sequence modeling with multi-interest representations is crucial for production recommendation, as it mitigates interest forgetting, strengthens cold-start robustness, and provides a higher-quality candidate pool for downstream ranking without sacrificing efficiency. As shown by these Tables, MuSeR establishes a new state of the art on both Recall@100 and Recall@500 across scenarios. These improvements confirm that capturing multi-scale temporal

TABLE II: Quantitative comparison on the Home Feed scenario. Best results are in bold.

Method	Recall@100	Recall@500
SASRec [27]	0.0389	0.1176
NANN [32]	0.0714	0.1982
TIGER [36]	0.0511	0.1654
KuaiFormer	0.0524	0.1701
Ours (MuSeR)	0.0870	0.2232

TABLE III: Quantitative comparison on the Discovery and Short Video scenario. Best results are in bold.

Method	Recall@100	Recall@500
SASRec [27]	0.0569	0.1372
NANN [32]	0.1012	0.2287
TIGER [36]	0.0721	0.1881
KuaiFormer	0.0818	0.1932
Ours (MuSeR)	0.1377	0.2534

patterns and disentangling user intents are essential for robust recall in real-world traffic, while multimodal semantic alignment enhances coverage for sparse and newly published items. The resulting candidate sets enable downstream re-rankers to operate on more relevant pools, ultimately contributing to better end-to-end personalization quality under real-world production constraints.

D. Ablation Study

To understand the contribution of each key component, we first examine different sequence processing strategies. As shown in Figure 4a, comparing long sequence without compression (0.2351), truncation to short sequence (0.1749), and our proposed hierarchical compression (0.2232) reveals several important insights. The significant performance drop with truncation (-0.0602) demonstrates the critical importance of preserving long-term behavioral signals. While compression introduces a slight performance reduction compared to uncompressed sequences (-0.0119), it enables practical deployment with an acceptable accuracy-efficiency trade-off. This finding validates our design choice of hierarchical compression as a balanced solution for industrial-scale deployment.

The impact of multimodal semantic alignment proves particularly striking in our ablation experiments. Figure 4b demonstrates substantial improvements when incorporating multimodal components, with Home Feed showing a 36.4% relative improvement (from 0.1636 to 0.2232) and Discovery demonstrating a 33.9% enhancement (from 0.1892 to 0.2534). These remarkable gains confirm our hypothesis that enriching sparse ID embeddings with semantic signals significantly enhances recommendation quality. The improvement is particularly pronounced in cross-modal scenarios like Discovery, where content understanding plays a crucial role in user preference modeling. In our evaluation of retrieval strategies, we compare three distinct approaches: Brute-Force search, Beam search, and HNSW. As illustrated in Figure 4c, while Brute-Force achieves the highest recall scores (Home: 0.2328, Discovery:

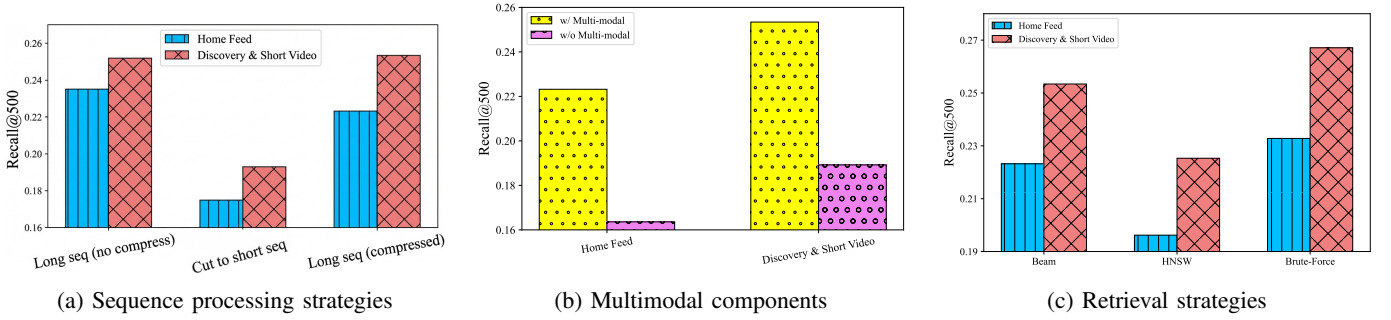


Fig. 4: Recsys study results.

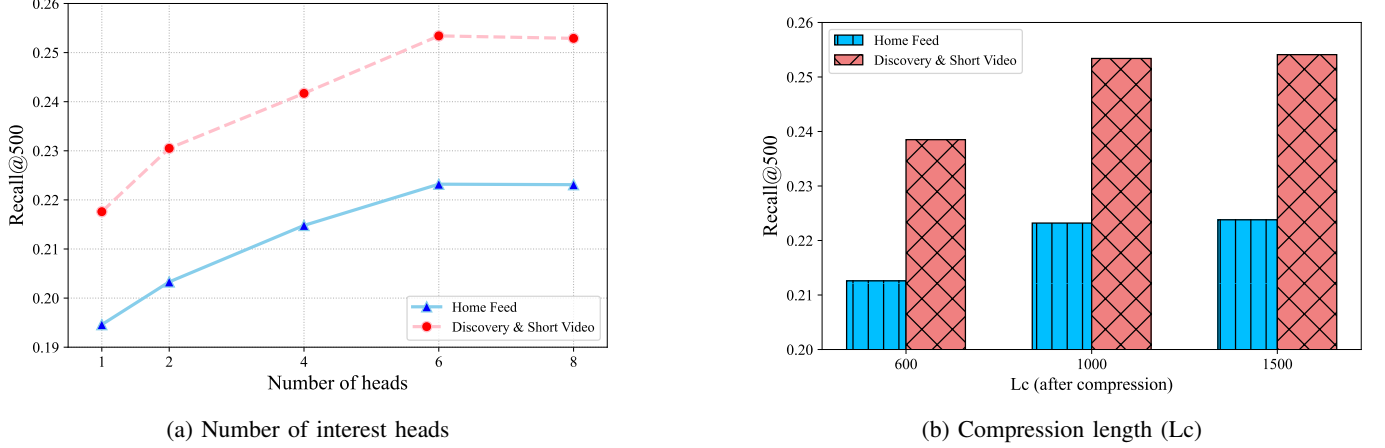


Fig. 5: Hyperparameter sensitivity analysis. (a) The optimal number of interest heads is around 4-6 for both scenarios, balancing model capacity and overfitting risks. (b) Compression length analysis indicates that $L_c=1000$ provides the best trade-off between information preservation and computational efficiency.

0.2671), it comes with prohibitive computational costs for industrial deployment. Beam search emerges as a particularly compelling solution, maintaining strong performance (Home: 0.2232, Discovery: 0.2534). HNSW offers the fastest retrieval but with a moderate sacrifice in overall recall quality.

E. Hyperparameter Analysis

The configuration of sequence length plays a crucial role in balancing model effectiveness and computational efficiency. We examine the impact of compressed sequence length by varying N_r (recent sequence) and N_m (mid-term sequence) while fixing Pool_k parameters ($k_1 = 16$, $k_2 = 64$). As shown in Figure 5b, increasing the compressed length from 600 to 1000 yields clear gains, with Home Feed improving from 0.2126 to 0.2232 and Discovery improving from 0.2385 to 0.2534. This indicates that moderately longer contexts still provide additional useful signals, especially for capturing recurring preferences and cross-session transitions. However, further extending the length to 1500 shows diminishing returns, with only 0.0006~0.0007 improvement, suggesting that the model reaches a performance plateau under the current compression granularity and serving constraints. Overall, the chosen configuration ($N_r = 800$, $N_m = 2000$) achieves robust performance across scenarios, reflecting a practical

“sweet spot” where longer histories improve effectiveness while keeping inference cost and latency manageable.

We further study the multi-interest head configuration. Figure 5a shows that multi-head mechanisms consistently outperform single-head baselines across all scenarios, confirming the importance of disentangling heterogeneous user intents. The best overall setting is 6 attention heads, achieving recall scores of 0.2232 for Home Feed and 0.2534 for Discovery. The Discovery scenario benefits more from multi-interest modeling than Home Feed, showing a larger relative improvement (0.0358 vs. 0.0286), which aligns with its higher content diversity and faster intent shifts. When the number of heads exceeds 6, performance plateaus or slightly decreases, implying that overly fine-grained interest decomposition can increase optimization difficulty and introduce redundancy. In practice, this suggests that a moderate number of heads provides the best trade-off between representation capacity and model complexity for stable online deployment.

V. RELATED WORKS

A. Sequence Modeling

Recent work tackles long-sequence recommendation through efficiency-oriented pipelines and scalable

architectures. Two-stage designs such as SIM [37] and TWIN [38] improve serving efficiency by first retrieving a target-relevant sub-sequence and then applying expressive attention over the filtered history; because the retained history is ultimately summarized into a single user representation, however, concurrent and conflicting interests remain only weakly separated. LONGER [2] scales Transformer-style modeling to substantially longer interaction sequences, yet heterogeneous interests and the gap between ID and content semantics are still difficult to reconcile under online latency constraints. Classical sequential models including GRU4Rec [26], SASRec [27], and BERT4Rec [28] target comparatively short sequences, and their purely ID-based representations offer limited leverage for long-range dependencies and for items or users whose interaction history is still sparse.

Multi-interest modeling has consequently become a major research direction, from MIND [39], which applies dynamic routing to extract multiple interest capsules from a behavior sequence, to controllable frameworks [40] that expose the number and granularity of extracted interest vectors as tunable factors. Subsequent efforts adopt contrastive objectives [41] to sharpen the boundaries between distinct interests, and target-aware distillation [42] to reconcile multi-interest training with single-vector inference at serving time. Related lines pursue disentanglement [43] to reduce redundancy across learned interests, capsule-based extraction [7], [44] to capture compositional preferences, and multi-granularity modeling [45], [46] to express intent at both coarse and fine levels, with query-aware methods forming a closely related thread [47], [48]. High-capacity architectures such as KuaiFormer [21] and HSTU [49] deliver strong accuracy but incur inference cost that is difficult to absorb in real-time retrieval. On the alignment side, VISTA [20] and BGE-M3 [50] yield stronger semantic representations, yet they are not designed for sequential recommendation and provide no mechanism for tracking how interests evolve over time. MuSeR instead couples hierarchical compression, multi-interest extraction, and semantic alignment in a single end-to-end framework, whereas prior work typically optimizes only one of these axes in isolation.

B. System Optimization Techniques

Industrial deployment must balance model accuracy against serving efficiency under stringent latency and throughput requirements. MARM [51] examines memory–compute trade-offs for caching intermediate representations under strict SLAs, but assumes a single cached state per user; multi-interest serving instead maintains several embeddings per user, whose refresh cadence should arguably differ according to how quickly each interest drifts. Asynchronous refresh in practice must also absorb traffic volatility and hardware heterogeneity to avoid latency spikes at peak load, a requirement that receives comparatively little explicit treatment. On the retrieval side, HNSW [15] supports fast approximate nearest-neighbor search with sub-linear query complexity, yet its fixed similarity metric can diverge from production ranking criteria,

particularly when multimodal document towers and multi-head user interests must be traded off jointly against relevance, diversity, and freshness. MuSeR addresses this through hierarchical beam search that re-scores candidate neighborhoods with learned multi-objective functions, and through asynchronous long- and short-term fusion with cache-aware refresh scheduling that adapts update intervals to monitored traffic, sustaining throughput under fluctuating QPS while preserving expressiveness within production latency budgets.

VI. CONCLUSION

We presented MuSeR, a scalable long-sequence recommendation framework that unifies hierarchical temporal compression, multi-query interest extraction, and multimodal semantic alignment in a production-ready retrieval pipeline. By preserving per-user histories of up to 10^5 interactions at multiple temporal resolutions, disentangling heterogeneous intents into distinct interest vectors, and enriching sparse IDs with textual semantics, MuSeR improves recall coverage, robustness to intent drift, and recommendation quality for recently published items. Experiments on public benchmarks and a large-scale Baidu dataset, together with online A/B tests, show statistically significant gains in feed DAU and total session duration while meeting production latency budgets. Future work will explore adaptive compression, dynamic head allocation, and retrieval–ranking co-design at billion-scale.

ACKNOWLEDGEMENT

This research was deployed in real-world business scenarios with the support of Baidu’s Feed Recall team and Recommendation Engine team. This research was supported by the Hong Kong Institute of AI for Science (HKAI-Sci), City University of Hong Kong. This research was also partially supported by the National Natural Science Foundation of China (No. 62502404), the Hong Kong Research Grants Council (Research Impact Fund No. R1015-23, Collaborative Research Fund No. C1043-24GF, RGC Research Fellow Scheme No. RFS2627-1S03, General Research Fund No. 11218325 and No. 11212926), and the Institute of Digital Medicine of City University of Hong Kong (No. 9229503).

REFERENCES

- [1] P. Covington, J. Adams, and E. Sargin, “Deep neural networks for youtube recommendations,” in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016.
- [2] Z. Chai, Q. Ren, X. Xiao *et al.*, “Longer: Scaling up long sequence modeling in industrial recommenders,” in *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, ser. RecSys ’25. New York, NY, USA: ACM, 2025, pp. 247–256, information for future conference, details may be subject to change.
- [3] L. Pan, W. Pan, M. Wei *et al.*, “A survey on sequential recommendation,” *arXiv preprint arXiv:2412.12770*, 2024.
- [4] X. Xu, H. Wang, W. Guo *et al.*, “Multi-granularity interest retrieval and refinement network for long-term user behavior modeling in ctr prediction,” in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, ser. KDD ’25. New York, NY, USA: ACM, 2025, pp. 2745–2755, information for future conference, details may be subject to change.
- [5] J. Lian, I. Batal, Z. Liu, A. Soni, E. Y. Kang, Y. Wang, and X. Xie, “Multi-interest-aware user modeling for large-scale sequential recommendations,” *arXiv preprint arXiv:2102.09211*, 2021.

- [6] Y. Zhang, W. He, L. Cui *et al.*, “Multi-interest aware recommendation in crowdintell network,” in *2020 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom)*. IEEE Computer Society, 2020, pp. 698–705.
- [7] Y. Cheng, Y. Fan, Y. Wang *et al.*, “Accurate multi-interest modeling for sequential recommendation with attention and distillation capsule network,” *Expert Systems with Applications*, vol. 243, p. 122887, 2024.
- [8] B.-C. Wu, Y.-L. Cheng, H.-S. Yuan *et al.*, “When multi-behavior meets multi-interest: Multi-behavior sequential recommendation with multi-interest self-supervised learning,” in *2024 IEEE 40th International Conference on Data Engineering (ICDE)*. IEEE, 2024, pp. 845–858.
- [9] Y. Meng, C. Guo, X. Hu, H. Deng, Y. Cao, T.-Y. Liu, and B. Zheng, “User long-term multi-interest retrieval model for recommendation,” in *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, ser. RecSys ’25. New York, NY, USA: ACM, September 2025, pp. 1112–1116, information for future conference, details may be subject to change.
- [10] J. Yan, L. Jiang, J. Cui *et al.*, “Trinity: Syncrizing multi-/long-tail/long-term interests all in one,” in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD ’24. New York, NY, USA: ACM, 2024, pp. 6095–6104.
- [11] Z. Wang and Y. Shen, “Incremental learning for multi-interest sequential recommendation,” in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 1071–1083.
- [12] F. Li, Q. Gao, Y. Dang *et al.*, “Denoising multi-interest-aware logical reasoning for long-sequence recommendation,” in *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’25. New York, NY, USA: ACM, 2025, pp. 1487–1496, information for future conference, details may be subject to change.
- [13] G. Zhou, N. Mou, Y. Fan, Q. Pi, W. Bian, C. Zhou, X. Zhu, and K. Gai, “Deep interest evolution network for click-through rate prediction,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 5941–5948.
- [14] A. Vaswani and *et al.*, “Attention is all you need,” *NeurIPS*, 2017.
- [15] Y. A. Malkov and D. A. Yashunin, “Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs,” *IEEE TPAMI*, 2018.
- [16] J. Ni, J. McAuley *et al.*, “The amazon 2023 review data: A contemporary benchmark for recommender systems,” in *Dataset/Report*, 2023.
- [17] McAuley-Lab, “Amazon-reviews-2023 dataset,” <https://huggingface.co/datasets/McAuley-Lab/Amazon-Reviews-2023>, 2023.
- [18] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, “Neural collaborative filtering,” in *WWW*, 2017.
- [19] S. Xiao *et al.*, “C-pack: Packaged resources to advance general chinese embedding,” in *arXiv:2307.09288*, 2023.
- [20] J. Zhou, Z. Liu, S. Xiao, B. Zhao, and Y. Xiong, “Vista: Visualized text embedding for universal multi-modal retrieval,” *arXiv preprint arXiv:2406.04292*, 2024.
- [21] C. Liu, J. Cao, R. Huang, K. Zheng, Q. Luo, K. Gai, and G. Zhou, “Kuaiformer: Transformer-based retrieval at kuaishou,” *arXiv preprint arXiv:2411.10057*, 2024.
- [22] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *ICLR*, 2019.
- [23] —, “Sgdr: Stochastic gradient descent with warm restarts,” *ICLR*, 2017.
- [24] R. Pascanu, T. Mikolov, and Y. Bengio, “On the difficulty of training recurrent neural networks,” in *ICML*, 2013.
- [25] J. Tang and K. Wang, “Personalized top-n sequential recommendation via convolutional sequence embedding,” in *WSDM*, 2018.
- [26] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, “Session-based recommendations with recurrent neural networks,” in *ICLR*, 2016.
- [27] W.-C. Kang and J. McAuley, “Self-attentive sequential recommendation,” in *ICDM*, 2018.
- [28] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, “Bert4rec: Sequential recommendation with bidirectional encoder representations from transformers,” in *CIKM*, 2019.
- [29] K. Zhou, H. Yu, W. X. Zhao, and J.-R. Wen, “Filter-enhanced mlp is all you need for sequential recommendation,” in *WWW*, 2022.
- [30] 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 *CIKM*, 2020.
- [31] Y. Zhu, S. Zhang, and *et al.*, “Tiger: Transferable interest graph embedding for recommendation,” in *WWW*, 2023.
- [32] J. Chen, Y. Chen, Y. Liu, Z. Zhao, Z. Liu, and J. Tang, “Nann: A neural-network-based approach to nearest neighbor search for recommender systems,” in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, pp. 167–177.
- [33] A. Paszke, S. Gross, F. Massa, and *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” in *NeurIPS*, 2019.
- [34] T. Wolf, L. Debut, V. Sanh, and *et al.*, “Transformers: State-of-the-art natural language processing,” in *EMNLP: System Demonstrations*, 2020.
- [35] A. Sergeev and M. Del Balso, “Horovod: Fast and easy distributed deep learning in tensorflow,” in *ArXiv:1802.05799*, 2018.
- [36] S. Rajput, D. Verma, A. Kumar, Y. Ge, S. Kumar, and A. Jain, “Tiger: A deep-learning-based recommender system for large-scale e-commerce platforms,” *arXiv preprint arXiv:2307.03789*, 2023.
- [37] Q. Pi, X. Zhu, G. Zhou, Y. Zhang, Z. Wang, L. Ren, Y. Fan, and K. Gai, “Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction,” in *CIKM*, 2020.
- [38] J. Chang, C. Zhang, Z. Fu, X. Zang, L. Guan, J. Lu, Y. Hui, D. Leng, Y. Niu, Y. Song, and K. Gai, “Twin: Two-stage interest network for lifelong user behavior modeling in ctr prediction at kuaishou,” in *KDD*, 2023.
- [39] C. Li, Z. Liu, M. Wu *et al.*, “Multi-interest network with dynamic routing for recommendation at tmall,” in *Proceedings of the 28th ACM international conference on information and knowledge management*. ACM, 2019, pp. 2615–2623.
- [40] Y. Cen, J. Zhang, X. Zou *et al.*, “Controllable multi-interest framework for recommendation,” in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*. ACM, 2020, pp. 2942–2951.
- [41] S. Zhang, L. Yang, D. Yao *et al.*, “Re4: Learning to re-contrast, re-attend, re-construct for multi-interest recommendation,” in *Proceedings of the ACM web conference 2022*. ACM, 2022, pp. 2216–2226.
- [42] C. Wang, Z. Wang, Y. Liu *et al.*, “Target interest distillation for multi-interest recommendation,” in *Proceedings of the 31st ACM international conference on information & knowledge management*. ACM, 2022, pp. 2007–2016.
- [43] Y. Du, Z. Wang, Z. Sun *et al.*, “Disentangled multi-interest representation learning for sequential recommendation,” in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD ’24. ACM, 2024, pp. 677–688.
- [44] Z. Tang, L. Wang, L. Zou *et al.*, “Towards multi-interest pre-training with sparse capsule network,” in *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*. ACM, 2023, pp. 311–320.
- [45] Y. Tian, J. Chang, Y. Niu *et al.*, “When multi-level meets multi-interest: A multi-grained neural model for sequential recommendation,” in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*. ACM, 2022, pp. 1632–1641.
- [46] C. Meng, Z. Zhao, W. Guo *et al.*, “Coarse-to-fine knowledge-enhanced multi-interest learning framework for multi-behavior recommendation,” *ACM Transactions on Information Systems*, vol. 42, no. 1, pp. 1–27, 2023.
- [47] T. Guo, X. Li, H. Yang *et al.*, “Query-dominant user interest network for large-scale search ranking,” in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. ACM, 2023, pp. 629–638.
- [48] R. W. White, P. N. Bennett, and S. T. Dumais, “Predicting short-term interests using activity-based search context,” in *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, 2010, pp. 1009–1018.
- [49] J. Zhai, L. Liao, X. Liu, Y. Wang, R. Li, X. Cao, L. Gao, F. Gu, M. He, Y. Lu, and Y. Shi, “Actions speak louder than words: Trillion-parameter sequential transducers for generative recommendations,” *arXiv preprint arXiv:2402.17152*, 2024.
- [50] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, “BGE m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation,” *arXiv preprint arXiv:2402.03216*, 2024.
- [51] X. Lv, J. Cao, S. Guan, X. Zhou, Z. Qi, Y. Zang, M. Li, B. Wang, K. Gai, and G. Zhou, “Marm: Unlocking the future of recommendation systems through memory augmentation and scalable complexity,” *arXiv preprint arXiv:2411.09425*, 2024.