Proceedings of CONF-SEML 2026 Symposium: Computational Analysis and Modeling in Complex Intelligent Systems
DOI: 10.54254/2755-2721/2026.GU33760

The Journey of Ultra-Long Sequence Recommendation Models: From Heuristic and Learnable Retrieval to Mathematical Architecture Optimization

Anbang Zhu

College of Computer Science, Zhejiang University, Hangzhou, China
3230103190@zju.edu.cn

Abstract. As user interactions in contemporary online platforms expand, sequential recommendation systems are encountering computational challenges. Users' past interaction sequences have grown from tens of items in the early days to life-long sequences with tens of thousands or millions of interactions. Despite the clear benefits of the self-attention mechanism embodied in the Transformer model in capturing long-term dependencies, the computational and memory costs of the self-attention mechanism scale quadratically $\left(O\left(L^2\right)\right)$ with the input sequence length, making it difficult to apply in real-time industry-level recommendation systems with high concurrency and low latency. This review provides an overview of the evolution of ultra-long sequence recommendation models, grouping the current solutions into three main paradigms: heuristic and learnable retrieval, state compression based on memory and hierarchical models, and mathematical architecture optimization based on linear attention and state space models (SSMs). This paper also investigates the recent developments in this area, examining the impact of the cloud-edge collaborative computing paradigm in combining computational resources and user privacy, the denoising strategies in interactive sequences and the potential use of large language models (LLMs) for semantic understanding and generation in long sequences. By examining the theoretical frameworks and experimental results of key works, this paper offers an outlook on future research directions for the next generation of recommendation systems with high scalability, privacy protection and computational efficiency.

Keywords: Ultra-long sequence recommendation, Linear attention, State space models, Cloud-edge collaboration, Sequence denoising

1. Introduction

The ability to mine users' ever-changing preferences from large-scale data has positioned Sequential Recommendation (SR) as a key component in e-commerce, streaming, and social media platforms. Traditional recommendation systems primarily adopted collaborative filtering or pattern matching, which usually considered users' interaction histories as unordered sets. With the advent of deep learning, Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) were frequently employed to model the dynamic changes of users' interests. Later, the advent of the

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.GU33760

Transformer architecture also boosted the development of sequential recommendation. The self-attention mechanism can capture long-range and local inter-item dependence, thus delivering remarkable accuracy gains.

But it leads to higher computational complexity with the length of the sequence. As user dwell time on platforms grows and the amount of interaction data increases, user behavior sequences gradually transform from short-term sessions to long-term sequences with tens of thousands of items. Under these circumstances, the shortcomings of the traditional Transformer architecture become more apparent: the time and space complexity of its self-attention mechanism is quadratic $\left(O\left(L^2\right)\right)$ with respect to the sequence length L [1]. For L in the tens of thousands, the calculation of the attention matrix consumes a large amount of GPU memory and increases the online inference time.

To resolve this issue, industry first resorted to a simple truncation, where only the last few dozen user interactions were taken as input to the model [2]. While this meets the strict requirements for system latency, it inevitably neglects long-term interests and regularity in a user's historical data. As a result, the industry and academia have been striving to build system frameworks to reduce the $O\left(L^2\right)$ complexity.

This paper intends to provide a sketch of the journey of ultra-long sequence recommendation. This paper shall discuss the development from the initial two-stage retrieval models to the compression of sequences through memory networks, and finally to mathematical reconstruction based on linear complexity and state space models. At the same time, this paper will discuss the explorations of cloud-edge collaborative recommendation, denoising in complex sequences, and long-context generative recommendation based on large language models.

2. Retrieval-driven sequence selection and pruning

A system engineering solution to address ultra-long sequences is the "Retrieve-then-Process" paradigm. The retrieval paradigm can reduce the length L of the sequence actually processed by the attention layer of the deep neural network by removing the low-relevance interactions before the data is fed into the deep neural network, therefore avoiding the quadratic computational complexity.

2.1. Hard retrieval with heuristic search and two-stage architectures

One early breakthrough is the Search-based Interest Model (SIM) [2], which achieved good results for lifetime sequences of up to 54, 000 events. SIM separates the modeling of a sequence into two phases: the General Search Unit (GSU) and the Exact Search Unit (ESU). GSU is a low-cost search unit that searches the user's historical data to retrieve a sub-sequence related to the target candidate item.

Originally, SIM's GSU mainly used some heuristic hard search rules, like category exact matching. Suppose the target candidate is a smartphone, the GSU will only extract interactions from the "electronics" category in the user's history. Then the ESU will focus on this sub-sequence through a target-aware attention mechanism. While SIM effectively reduces the sequence length to a manageable size, the hard search mechanism has its drawbacks: consumer interests can be cross-domain, and hard matching easily excludes cross-domain interest, resulting in important information loss.

DOI: 10.54254/2755-2721/2026.GU33760

2.2. End-to-end learnable retrieval mechanisms

To address the drawbacks of heuristic rules, a learnable retrieval strategy was proposed. User Behavior Retrieval for Click-Through Rate prediction (UBR4CTR) [3] introduced an end-to-end learnable retrieval mechanism, where the retrieval process is also considered a trainable module. UBR4CTR no longer uses static category matching, but builds a query vector by combining the target item, target user, and other contextual features.

Furthermore, by training the retrieval process with the downstream click-through rate prediction, UBR4CTR can dynamically choose historical behaviors. The model is trained to find the best match for the CTR prediction task, keeping cross-domain signals that may be lost with hard matching. This guarantees long-term and periodic patterns are actively searched.

2.3. Hash-based approximation and acceleration

While learnable retrieval boosts semantic fidelity, dense vector inner products for scoring millions of interactive candidates in ultra-long cases still remains expensive. To further accelerate performance, methods based on probabilistic approximation, such as Locality Sensitive Hashing (LSH) were developed.

The End-to-end Target Attention (ETA) model [4] adopts SimHash to reduce the dimensionality of the item embedding vector to a binary hash feature vector. ETA transforms continuous vector spaces into binary spaces, and performs XOR bitwise computations rather than dot product operations to calculate the Hamming distance. This kind of hash-based sampling enables the system to sample the relevance of a large number of historical items in milliseconds.

Hash-based retrieval was further improved in the Sampling-based Deep Interest Modeling (SDIM) architecture [5]. SDIM uses multiple independent hash functions in the retrieval process to divide historical items into hash buckets; if a historical item and the target item are in the same hash bucket under multiple hash functions, it is sampled. As such, SDIM achieves a similar effect to full-sequence target attention with a linear time complexity of $O(L)$. As shown in Table 1, such retrieval paradigms show an evolution towards solving the efficiency-effectiveness challenge.

Table 1. Summary of major core retrieval methods for ultra-long sequence recommendation

Model	Retrieval Mechanism	Core Technology	Complexity/ Acceleration Strategy	Limitations
SIM	Heuristic hard retrieval	Category-based inverted index and exact matching	Avoids $O(L^2)$ global attention, instead uses local sub-sequence computation	Information loss due to losing cross-domain interests that may not be included in the same category
UBR4CTR	Learnable soft retrieval	End-to-end jointly optimized query vector retrieval	Dynamic sub-sequence filtering based on the target item	Dense vector dot products still suffer from the computational issue at million-scale lengths
ETA	Hash approximate retrieval	SimHash binary signatures and Hamming distance	Bitwise operations (XOR) replace matrix dot products, achieving $O(L^2)$ acceleration	Dimensionality reduction hashing may lead to loss of precision and hash collisions

DOI: 10.54254/2755-2721/2026.GU33760

Table 1. (continued)

SDIM	Probabilistic local sampling	Multi-hash function bucketing and locality-sensitive collision	Removes attention weight computation, achieving linear complexity	Depends on the quality of hash functions and their adaptability to data distributions

3. Memory-based state compression

The retrieval paradigm is mainly built on the assumption that most of the past information is not relevant to the current target. However, users' long-term interests are constantly building up. The sequence compression paradigm moves away from data filtering to summarisation, with the goal of compressing sequences into mathematical representations.

3.1. Memory-based networks and saturation

A fundamental way to compress sequences is to integrate external memory modules, thus separating the modelling of user interests from the size of sequences. The multi-channel user Interest Memory Network (MIMN) [6] is inspired by the Neural Turing Machines (NTMs), which stores a fixed-size external memory matrix as the user's global interests.

Upon each new interaction, the model can update the matrix at particular slots using read/write heads. This design allows the model to perform incremental state updates in a constant $O(1)$ time. But MIMN and its variants suffer from the problem of "representation saturation. " The network needs to update the memory matrix with a limited number of memory slots by overwriting previous data with new data, which could cause long-term subtle preferences to be diluted by recent frequent activities.

3.2. Multi-scale hierarchical interest modeling

To reduce the representation problem and to separate long-term interests and short-term intentions, models such as Hierarchical Network with User Memory (HNUM) [7] have been developed.

HNUM has a two-layer routing scheme. The bottom layer is a session-level Gated Recurrent Unit (GRU) that receives the output of the top-layer user-level memory and extracts the short-term intention changes by modelling the transitions between items within the ongoing active session. The upper layer is a user-level memory network that takes the output of the bottom-layer session encoder and updates it to a fixed state. By using such a compression mechanism, HNUM increases the efficiency of the top-level attention mechanism by reducing the number of tokens to attend to, while maintaining accuracy.

4. Optimization: from linear to state space models

While memory compression and retrieval have been successful in engineering practice, recent research is aimed at re-defining the mathematical structure of sequence modeling operators to further improve their effectiveness and efficiency.

DOI: 10.54254/2755-2721/2026.GU33760

4.1. L2 normalization and linear attention mechanisms

The quadratic $O(L^2)$ complexity of the standard Transformer architecture is largely due to the Softmax normalization performed on the dot product of the Query (Q)matrix and Key (K) matrix. In the original attention operation $V' = Softmax(QK^T)V$ the associative property of matrix multiplication is violated due to the non-linearity of Softmax, making it necessary to materialise the $L \times L$ attention matrix.

This is improved by the L2-normalized Linear Attention Mechanism for Long-term Sequential Recommender Systems (LinRec) [7]. LinRec eliminates the Softmax coupling, row-wise L2-normalized Q matrix, column-wise L2-normalized K matrix, and a non-linear activation function (e.g., ELU layer).

The removal of Softmax coupling allow the computation formula $(QK^T)V$ to be equivalently rewritten as $Q(K^TV)$. The dimension of K^T is $d \times L$ and V is $L \times d$ (usually $d \ll L$), K^TV forms a $d \times d$ matrix. Then multiply Q with it, the total complexity is reduced to $O(Ld^2)$. When the sequence length L is much greater than the hidden layer size d , it has a linear complexity $O(L)$.

4.2. Parallelism of linear recurrent units

An alternative architecture for long sequence recommendation is the Linear Recurrent Units (LRUs). The LRURec model [8] uses LRUs to model time-dependency and simplifies it to real-valued parameters to align with the discrete nature of recommendation data.

The LRURec architecture takes the nonlinear activation function outside the recurrent state update, resulting in a linear state transition equation. This enables the application of parallel scan algorithms for forward propagation computation on the time axis, enhancing the training process. During inference, LRURec has $O(1)$ state updates and $O(L)$ inference costs.

4.3. State space models (SSMs)

Discrete-time State Space Models (SSMs) originated from continuous-time control theory have been demonstrated to be useful in sequential tasks.

The Mamba for recommendation model (Mamba4Rec) [9] combines selective scan with the state space approach. Mamba4Rec makes state transition parameters input-dependent, enabling the model to selectively decide on whether to keep the valuable information in the hidden state based on the inputs, and discard the useless information. Moreover, Mamba4Rec adopts hardware-efficient parallel prefix sum operations to improve the computational efficiency and achieve near real-time inference for sequential data.

5. State-of-the-art: cloud-edge collaboration, denoising and LLMs

As models operate on an increasing number of items, industrial applications need to consider additional constraints such as privacy, noise and semantics.

DOI: 10.54254/2755-2721/2026.GU33760

5.1. Real-time computation with privacy protection under cloud-edge collaboration

When dealing with large data, conventional centralized models suffer from network bottleneck and privacy issues. The Cloud-Device Collaborative Agents for Sequential Recommendation (CDA4Rec) framework [10] addresses this issue through collaborative agents.

CDA4Rec breaks down the recommendation process into subtasks. The cloud server hosts a Large Language Model (LLM) to provide coarse-grained semantic comprehension and retrieve candidates from a global pool. The edge device deploys a Small Language Model (SLM) for structured user profiling and user-specific ranking. Any interaction data is processed on the device, avoiding network leakage.

The SLM on the device creates a "User Abstract" without identity information. The cloud LLM, a strategy planner, receives and processes the abstract to produce an execution plan. CDA4Rec enables parallel execution of the cloud and device to enhance privacy and speed up the process.

5.2. Denoising for interactive sequences

Long sequences may contain noise due to accidental clicks or meaningless explorations. Linear models or state space models that process long sequences can accumulate noise.

The Interest Alignment for Denoising Sequential Recommendation (IADSR) approach [11] offers a multimodal method for noise recognition. IADSR employs a sequence encoder to obtain collaborative embeddings and an LLM to obtain semantic embeddings. It then measures the geometric distance between the collaborative modality and the semantic modality, and identifies interactions with large distance as noise and removes them. This approach helps solve the "over-denoising" issue that arises from solely using collaborative filtering signals.

Beyond this, the LLM4DSR framework [12] also leverages the reasoning ability in LLMs for denoising. This approach designs a self-supervised fine-tuning task to trigger the capability of LLM to detect noisy outliers. At the same time, its inherent uncertainty estimation module measures the confidence level of the judgement, ensuring the correction operation is only applied when the confidence is above a certain threshold to avoid removing important information. As demonstrated in Table 2, these cutting-edge systems offer effective approaches for complex sequences.

Table 2. Core design comparison of ultra-long sequences denoising and collaborative mechanisms

Model / Framework	Core Objective	Algorithm Mechanism & Technical Path	Key Problem Addressed	Performance & Deployment Benefits
CDA4Rec	Cloud-edge privacy collaboration	Dynamic strategy prompting & dual-agent (LLM/SLM) decoupled execution	Dilemma between privacy risk of sensitive data transmission and huge computing resources	Latency<100ms, personalized ranking with privacy isolation
IADSR	Semantic collaborative denoising	Geometric divergence alignment between CF space and LLM semantic space	Over-denoising of long-tail items caused by traditional frequency filtering rules	Reduces recursive accumulation of noise errors in linear state space models
LLM4DSR	Anomaly detection denoising	Self-supervised fine-tuning for anomaly recognition & uncertainty estimation	Logical inconsistencies caused by LLMs directly handling ID sequences	Enables feature purification without needing additional labels

DOI: 10.54254/2755-2721/2026.GU33760

5.3. Large language model-driven long-context recommendation paradigm

Generative recommendation models have the potential to support transfer learning across different domains and solve cold-start issues by translating discrete item IDs (identifiers) or textual attributes into semantic tokens.

The Hierarchical Sequential Transduction Units (HSTU) [13] is one such engineering solution to extend generative models. HSTU, with hundreds of billion parameters, uses pointwise attention-aggregation and element-wise gated transformations to reduce the amount of computation. HSTU tackles variable interaction sequence length by introducing jagged tensor and hardware optimizations. This low-level engineering optimisation delivers a significant improvement in the maximum sequence length of HSTU.

Moreover,the VIrtual Sequential Target Attention (VISTA) framework [14] decouples interaction sequence processing system-level. VISTA provides an asynchronous pre-computation pipeline to generate a few "summarization tokens" from the historical sequences, storing them in a cache. At serving time, only the candidate items have to compute attention with the cached tokens, instead of the raw sequences. This decoupled approach successfully integrates long-context theories with the industry timeliness constraints.

6. Conclusion

The design of sequential recommendation systems has evolved from time recording to the understanding of lifelong sequences. The quadratic computational complexity introduced by the Softmax mechanism in early Transformer models lead to the industry applying truncation or retrieval techniques. Then, with learnable hashing and hierarchical memory network compression, the sequence processing efficiency was further enhanced.

Most recent developments focus on mathematical layers. The addition of L2-normalized linear attention layers and selective state space models theoretically have linear complexity ($O(L)$) and hardware efficiency. These models solve the representation saturation issue and can safely update global user representations.

In the future, the tight integration of generative recommendation systems with large language models suggests that sequence modeling is evolving from ID pattern matching to semantic reasoning. Tensor optimizations (HSTU) and asynchronous caching (VISTA) offer engineering solutions for large-scale systems. At the same time, cloud-edge collaborative frameworks such as CDA4Rec strike a balance between privacy and resource utilization in physical infrastructure. Together with denoising and distributed networks, future recommendation systems will have enhanced overall performance in responsiveness, accuracy and security.

References

[1] Liu, L., Cai, L., Zhang, C., Zhao, X., Gao, J., Wang, W., Lv, Y., Fan, W., Wang, Y., He, M., Liu, Z., & Li, Q. (2023). LinRec: Linear attention mechanism for long-term sequential recommender systems. Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 289–299. https: //doi.org/10.1145/3539618.3591717

[2] Pi, Q., Bian, W., Zhou, G., Zhu, X., & Gai, K. (2020). Search-based user interest modeling with lifelong sequential behavior data for click-through rate prediction. Proceedings of the 29th ACM International Conference on Information & Knowledge Management, 2685–2692.https: //doi.org/10.1145/3340531.3412744

[3] Qin, J., Zhang, W., Wu, X., Jin, J., Fang, Y., & Yu, Y. (2020). User behavior retrieval for click-through rate prediction. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2347–2356.https: //doi.org/10.1145/3397271.3401440

DOI: 10.54254/2755-2721/2026.GU33760

[4] Chen, Q., Xu, Y., Pei, C., Lv, S., Zhuang, T., & Ge, J. (2021). Efficient long sequential user data modeling for click-through rate prediction. arXiv preprint, arXiv: 2209.12212.
[5] Cao, Y., Zhou, X., Feng, J., Huang, P., Xiao, Y., Chen, D., & Chen, S. (2022). Sampling is all you need on modeling long-term user behaviors for CTR prediction. Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2974–2983.https: //doi.org/10.1145/3511808.3557083
[6] Pi, Q., Bian, W., Zhou, G., Zhu, X., & Gai, K. (2019). Practice on long sequential user behavior modeling for click-through rate prediction. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2671–2679. https: //doi.org/10.1145/3292500.3330673
[7] Dong, J., Sun, F., Wu, T., Wu, X., Zhang, W., & Wang, S. (2022). A hierarchical network with user memory matrix for long sequence recommendation. Wireless Communications and Mobile Computing, 2022(1), 5457044. https: //doi.org/10.1155/2022/5457044
[8] Yue, Z., Wang, Y., He, Z., Zeng, H., McAuley, J., & Wang, D. (2024). Linear recurrent units for sequential recommendation. Proceedings of the 17th ACM International Conference on Web Search and Data Mining, 930–938. https: //doi.org/10.1145/3616855.3635799
[9] Liu, C., Lin, J., Wang, J., Liu, H., & Caverlee, J. (2024). Mamba4Rec: Towards efficient sequential recommendation with selective state space models. arXiv preprint, arXiv: 2403.03900.
[10] Long, J., Huang, S., Huo, H., & Xu, G. (2025). CDA4Rec: Cloud-device collaborative agents for sequential recommendation. arXiv preprint, arXiv: 2509.01551.
[11] Wu, T., Huang, C., Huang, H., Yu, T., & Xie, K. (2025). Empowering denoising sequential recommendation with large language model embeddings. arXiv preprint, arXiv: 2510.04239.
[12] Wang, B., Liu, F., Zhang, C., Chen, J., Wu, Y., Zhou, S., Lou, X., Wang, J., Feng, Y., Chen, C., & Wang, C. (2024). LLM4DSR: Leveraging large language model for denoising sequential recommendation. arXiv preprint, arXiv: 2408.08208.
[13] Zhai, J., Li, L., Liu, X., Wang, Y., & Li, R. (2024). HSTU: Hierarchical sequential transduction units for generative recommendation. Proceedings of the 41st International Conference on Machine Learning, PMLR 235.
[14] YChen, Z., Zhao, C., Mo, K. C., Jiang, Y., Lee, J. H., Mahajan, K. C., Jiang, N., Ren, K., Li, J., & Yang, W. Y. (2025). Massive memorization with hundreds of trillions of parameters for sequential transducer generative recommenders. arXiv preprint, arXiv: 2510.22049.