Hyperbolic-Enhanced Mixture-of-Experts Mamba for Sequential Recommendation

Yuwen Liu[1,2], Lianyong Qi[1,2*], Xingyuan Mao[1,2], Weiming Liu[3], Xuhui Fan[4], Qiang Ni[5], Xuyun Zhang[4], Yang Zhang[6*], Yuan Tian[7], Amin Beheshti[4]

[1]College of Computer Science and Technology, China University of Petroleum (East China), China
[2]Shandong Key Laboratory of Intelligent Oil and Gas Industrial Software, China
[3]ByteDance Inc., Singapore
[4]School of Computing, Macquarie University, Australia
[5]School of Computing and Communications, Lancaster University, UK.
[6]Anuradha and Vikas Sinha Department of Data Science, University of North Texas, USA
[7]School of Computer Engineering, Nanjing Institute of Technology, China
yuwenliu97@gmail.com, lianyongqi@upc.edu.cn, {hsingyuanmao,lwming95}@gmail.com, xuhui.fan@mq.edu.au, q.ni@lancaster.ac.uk, xuyun.zhang@mq.edu.au, yang.zhang@unt.edu, ytian@njit.edu.cn, amin.beheshti@mq.edu.au

Abstract

Sequential recommendation has emerged as a fundamental task in various domains, aiming to predict a user's next interaction based on historical behavior. Recent advances in deep sequence models, particularly Transformer-based architectures and the more recent Mamba, have substantially pushed the boundaries of sequential modeling performance. However, existing methods still face two critical challenges. First, many current approaches overlook the hierarchical structures and high-order dependencies among items, typically restricting representation learning to conventional Euclidean spaces, which limits their capacity to capture complex relational information. Second, although Mamba excels at long-range dependency modeling, its reliance on static Feed-Forward Networks (FFNs) hinders its ability to dynamically adapt to evolving user preferences across diverse contexts. To address these limitations, we propose a Hyperbolic-Enhanced Mixture-of-Experts Mamba recommender (HM2Rec) for sequential recommendation. HM2Rec first encodes user-item relationships through hyperbolic graph convolution to exploit hierarchical structure more effectively. Then, a Variational Graph Auto-Encoder (VGAE) is employed to reconstruct node embeddings, improving structural robustness. To further enhance sequential modeling, we integrate Rotary Positional Encoding (RoPE) into Mamba to better capture relative position dependencies, and replace the FFN with Mixture-of-Expert (MOE) module, enabling dynamic and personalized expert selection for each token. Our extensive experiments on four widely-used public datasets demonstrate that HM2Rec outperforms several advanced baseline models.

Introduction

Sequential recommendation aims to predict a user's next interaction from historical behaviors and has been widely studied across domains (Liu, Xia, and Huang 2024; Rajput et al. 2023). Early approaches (Donkers, Loepp, and Ziegler 2017; Bach, Long, and Phuong 2020) proposed to use Recurrent Neural Networks (RNNs) to capture temporal dependencies in interaction sequences. While effective in modeling short-term transitions, RNN-based methods suffer from difficulty in learning long-range dependencies and are prone to gradient vanishing issues. Graph Neural Networks (GNNs) (Ma et al. 2020; Peintner 2023) were introduced to model the intricate user-item interactions by treating interactions as nodes within a graph structure. However, traditional GNNs often struggle to capture sequential patterns explicitly and are limited by their shallow aggregation schemes. LightGCN (He et al. 2020) simplified GNN architectures by removing feature transformations and nonlinearities, achieving more efficient and scalable recommendation models; nevertheless, it mainly focuses on global collaborative signals and lacks the capability to model fine-grained sequential dynamics.

With the rapid development of deep learning techniques (Sun et al. 2023; Wang et al. 2025; Sun, Chen, and Yang 2019), neural architectures have demonstrated remarkable capabilities in representation learning and sequence modeling. Transformer-based methods (Yang et al. 2022; Chen et al. 2023) have been widely adopted in sequential recommendation, benefiting from self-attention mechanisms to capture complex dependencies across user interaction histories. However, Transformers typically demand heavy computational resources and may not effectively model long sequences without significant overhead. Recently, Mamba-based models (Liu et al. 2024a; Qu et al. 2024; Fan et al. 2024; Liu et al. 2025b) have achieved state-of-the-art performance in sequential recommendation. They mainly focus on architectural improvements while overlooking the rich prior knowledge embedded in interactions. Despite these advances, existing methods still face in particular two key limitations: (i) they often overlook the hierarchical and structural relationships among items during feature extraction, and (ii) even in state-of-the-art models like Mamba, the reliance on static feed-forward networks (FFNs) limits the flexibility needed to dynamically model diverse and evolving user behaviors.

*Corresponding author.

To address the above challenges, we propose a **H**yperbolic-Enhanced **M**ixture-of-Experts **M**amba **Rec**ommender (**HM2Rec**) for sequential recommendation, which simultaneously enhances feature representation and dynamic sequence modeling. Specifically, we construct an interaction graph and employ hyperbolic GCNs to embed nodes in a non-Euclidean space, naturally preserving hierarchical structures. To further improve the robustness of the learned representations, we apply a variational reconstruction process that refines the graph structure and enhances the quality of node embeddings. Based on the hyperbolic-enhanced representations, we design a customized sequence model by integrating a Mixture-of-Experts (MOE) module into the Mamba backbone, replacing the standard FFN. Furthermore, we incorporate Rotary Positional Encoding (RoPE) with Mamba (RPMamba) to better capture relative positional dependencies across user interaction sequences. These designs enable the model to dynamically adapt its transformation functions based on the characteristics of different sequence contexts, thereby capturing diverse and evolving user preferences more effectively. Extensive experiments on four public datasets demonstrate that HM2Rec outperforms baselines, highlighting the effectiveness of both hyperbolic feature enhancement and MOE-based sequence modeling. In summary, the main contributions of this paper are: (1) We introduce a hyperbolic representation learning framework to model hierarchical user-item relationships using hyperbolic GCNs and variational graph reconstruction. (2) We propose a MOE-Augmented RPMamba architecture by replacing its feed-forward module with a Mixture-of-Experts layer and augmenting it with RoPE for improved sequential dependency modeling. (3) We conduct extensive experiments on four real-world datasets, and the results consistently validate the superiority of our proposed HM2Rec model against strong baselines.

Related Work

Hyperbolic-based Recommendation Modeling hierarchical structures is essential for recommendation, but Euclidean spaces often fail to capture such complexity (Liao et al. 2023; Liu et al. 2025a). Hyperbolic spaces, with their suitability for representing hierarchies, have thus gained traction in recommendation (Li et al. 2021; Zhang et al. 2022). Nickel and Kiela (Nickel and Kiela 2017) first introduce Poincar embeddings, demonstrating the effectiveness of hyperbolic geometry in modeling hierarchical structures. Following this, research efforts extend hyperbolic embeddings into recommendation scenarios. For instance, Chami et al. (Chami et al. 2019) propose Hyperbolic Graph Convolutional Neural Networks (HGCN), which adapt GNNs into hyperbolic spaces to better preserve the hierarchical relations among nodes. HGCH (Zhang and Wu 2024) and HGSR (Yang et al. 2023) leverage hyperbolic graph concepts: HGCH integrates side information into a heterogeneous graph, while HGSR exploits social structures via hyperbolic pre-training. HDRM(Yuan et al. 2025) uses a hyperbolic latent diffusion process for users and items, and by leveraging hyperbolic geometry for structural constraints. Despite these advances, most existing hyperbolic recommendation methods primarily focus on static settings or global structures. Few works have explored the potential of hyperbolic representations in enhancing dynamic, sequential recommendation tasks.

Mamba-based Recommendation Early methods (Donkers, Loepp, and Ziegler 2017; Bach, Long, and Phuong 2020; Tan, Xu, and Liu 2016) effectively capture short-term transitions but often suffer from limitations in modeling long-range dependencies. Transformer-based models (Kang and McAuley 2018; Sun et al. 2019; Fan et al. 2022; Yang et al. 2022; Chen et al. 2023; Liu et al. 2024b) were introduced to address these issues. While above methods effectively capture long-range dependencies, they often suffer from high computational cost and overlook structural priors.

Recently, Mamba (Gu and Dao 2024) has emerged as a promising sequence modeling framework, offering a lightweight and efficient alternative to traditional Transformer architectures. The strong modeling capacity and efficiency of Mamba have attracted attention in the recommendation community. For instance, Mamba4Rec (Liu et al. 2024a) first explore the integration of Mamba into sequential recommendation frameworks, aiming to leverage its ability to model long-range dependencies while maintaining computational efficiency. SSD4Rec (Qu et al. 2024) exploit a Bi-SSD block with variable-length sequences to capture user preferences for accurate recommendations. TiM4Rec (Fan et al. 2024) further propose a time aware enhancement method for Mamba architecture, aiming to reduce the performance loss of SSD through a time aware masking matrix. GeoMamba (Chen, Wang, and Shang 2024) present a Mamba-based geography encoder, treating item exact location representation as a sequence modeling task, efficiently modeling geographical features and sequential dependencies. These methods improve efficiency via state-space formulation but still ignore user-item structural priors and lack dynamic expert modeling.

Methodology

Preliminaries. Let $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$ represent the set of all users, and $\mathcal{V} = \{v_1, v_2, ..., v_{|\mathcal{V}|}\}$ denote the set of all items. The total number of users and items are $|\mathcal{U}|$ and $|\mathcal{V}|$, respectively. For each user $u \in \mathcal{U}$, we denote their historical check-in sequence as $S^u = \{s_1^u, s_2^u, ..., s_t^u\}$, where $s_t^u \in \mathcal{V}$ refers to the item visited at the t-th timestamp. The length of this sequence is represented by l_u, i.e., $|S^u| = l_u$. The goal of sequential recommendation is to predict the next item $s_{l_u+1}^u$ that user u is most likely to visit, given their current sequence S^u. And we present our HM2Rec framework whose overall architecture is shown in Figure 1.

To incorporate high-order item dependencies, we construct a item transition graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ using all users' interaction histories. In this graph, nodes correspond to items, and edges reflect the transitions derived from user trajectories. The graph's adjacency matrix $A \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ encodes the presence of transitions between items, where $a_{ij} = 1$ indicates a transition from p_i to p_j, and $a_{ij} = 0$ otherwise. The corresponding degree matrix is denoted as $D \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$. To define the edge set $\mathcal{E}$, we traverse each user's check-in se-

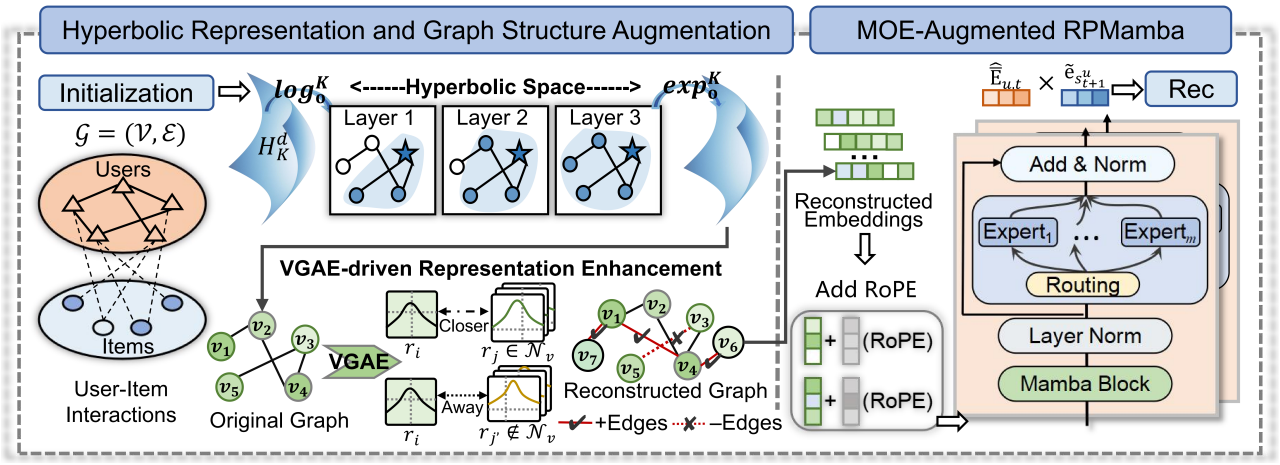

Figure 1: Overall architecture of our proposed HM2Rec.

quence and link each item to its n-hop neighbors within the same sequence. Specifically, the construction of $\mathcal{E}$ follows:

$$\mathcal{E} = \left\{ \left(s_i^u, s_j^u \right) : u \in \mathcal{U}, |i-j| \leq n, 1 \leq i < j \leq l_u \right\}. \tag{1}$$

Hyperbolic Representation and Graph Structure Augmentation

Hyperboloid Manifold. Let $(\mathbf{v}_i^{0,E})_{i \in \mathcal{V}}$ of size $\mathbb{R}^d$ denote the input features of item nodes in the Euclidean space, where the superscript ^{0} indicates the first layer, and the superscript ^{E} represents Euclidean embeddings. In contrast, we use the superscript ^{H} to indicate hyperbolic features. The Minkowski inner product is defined as: $\langle \mathbf{v}, \mathbf{x} \rangle_{\mathcal{L}} := -v_0 x_0 + \sum_{i=1}^{d} v_i x_i$, where $\langle \cdot, \cdot \rangle_{\mathcal{L}} : \mathbb{R}^{d+1} \times \mathbb{R}^{d+1} \to \mathbb{R}$ is the Minkowski bilinear form. The d-dimensional hyperboloid manifold with constant negative curvature $-1/K$ $(K > 0)$ is defined as: $\mathbb{H}_K^d := \left\{ \mathbf{v} \in \mathbb{R}^{d+1} : \langle \mathbf{v}, \mathbf{v} \rangle_{\mathcal{L}} = -K,\ v_0 > 0 \right\}$. Its corresponding tangent space at a point $\mathbf{v}$ is denoted as: $\mathcal{T}_{\mathbf{v}} \mathbb{H}_K^d := \left\{ \mathbf{y} \in \mathbb{R}^{d+1} : \langle \mathbf{y}, \mathbf{y} \rangle_{\mathcal{L}} = 0 \right\}$. The metric tensor is given by $g_{\mathcal{L}} = \mathrm{diag}[-1, 1, 1, ..., 1]$. Since there is no notion of vector space structure in hyperbolic spaces, it is necessary to implement the derive transformations in hyperbolic models. Specifically, we utilize the exp and log maps to implement Euclidean transformations in the target space $\mathcal{T}_{\mathrm{o}} \mathbb{H}^{d,K}$. Given $\mathbf{v} \in \mathbb{H}_K^d$, $\mathbf{y} \in \mathcal{T}_{\mathbf{v}} \mathbb{H}_K^d$ with $\mathbf{y} \neq 0$, and $\mathbf{x} \in \mathbb{H}_K^d$ such that $\mathbf{x} \neq \mathbf{v}$, the exp and log maps are defined as:

$$\exp_{\mathbf{v}}^K(\mathbf{y}) = \cosh\left(\frac{\|\mathbf{y}\|_{\mathcal{L}}}{\sqrt{K}} \right) \mathbf{y} + \sqrt{K} \sinh\left(\frac{\|\mathbf{v}\|_{\mathcal{L}}}{\sqrt{K}} \right) \frac{\mathbf{y}}{\|\mathbf{y}\|_{\mathcal{L}}}, \tag{2}$$

$$\log_{\mathbf{v}}^K(\mathbf{x}) = d_{\mathcal{L}}^K(\mathbf{v}, \mathbf{x}) \cdot \frac{\mathbf{x} + \frac{1}{K} \langle \mathbf{v}, \mathbf{x} \rangle_{\mathcal{L}} \mathbf{v}}{\left\| \mathbf{x} + \frac{1}{K} \langle \mathbf{v}, \mathbf{x} \rangle_{\mathcal{L}} \mathbf{v} \right\|_{\mathcal{L}}}, \tag{3}$$

where $\|\mathbf{y}\|_{\mathcal{L}} = \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle_{\mathcal{L}}}$ is the Lorentzian norm, and the hyperbolic distance function is given by:

$$d_{\mathcal{L}}^K(\mathbf{v}, \mathbf{x}) = \sqrt{K} \operatorname{arcosh}\left(-\langle \mathbf{v}, \mathbf{x} \rangle_{\mathcal{L}} / K \right). \tag{4}$$

Hyperbolic Initialization Layer. To embed the initial item representations into hyperbolic space, we begin by projecting the Euclidean embeddings into the hyperboloid manifold $\mathbb{H}_K^d$. Let $\mathbf{o} := \{\sqrt{K}, 0, ..., 0\} \in \mathbb{H}_K^d$ denote the origin of the manifold, which serves as the base point for tangent space mappings. Each item's hyperbolic initialization $\mathbf{v}^{0,H} \in \mathbb{H}_K^d$ is obtained by applying the exponential map centered at $\mathbf{o}$:

$$\mathbf{v}^{0,H} = \exp_{\mathbf{o}}^K(\mathbf{v}^{0,\mathcal{T}}), \tag{5}$$

where the tangent vector $\mathbf{v}^{0,\mathcal{T}} = (0, \mathbf{v}^{0,E})$ lies in the tangent space $\mathcal{T}_{\mathbf{o}} \mathbb{H}_K^d$, and $\mathbf{v}^{0,E} \in \mathbb{R}^d$ is sampled from a multivariate Gaussian distribution. Here, the prefix zero in $\mathbf{v}^{0,E}$ indicates the initial layer, while the superscript $^{\mathcal{T}}$ specifies that the vector lies in the tangent space at the origin. So we can interpret $(0, \mathrm{v}^{0,E})$ as a point in $\mathcal{T}_{\mathrm{o}} \mathbb{H}_K^d$ and map it to $\mathbb{H}_K^d$.

Feature Aggregation in Hyperbolic Space. Prior work (He et al. 2020) has shown that the inclusion of feature transformation layers and non-linear activation functions within graph convolution is not always beneficial and may even hinder performance. Motivated by this insight, we similarly omit these components in our hyperbolic graph aggregation process to preserve simplicity and computational efficiency. To perform feature aggregation in hyperbolic space, we leverage the exponential and logarithmic maps, which allow transformation between the manifold and its tangent space. Initially, each hyperbolic embedding $\mathbf{v}^{0,H}$ is projected into the tangent space at the origin $\mathbf{o}$ through the logarithmic map:

$$\mathbf{v}^{0,\mathcal{T}} = \log_{\mathbf{o}}^K(\mathbf{v}^{0,H}). \tag{6}$$

Given a hyperbolic graph $\mathcal{G}$ that captures relational transitions between items, we denote $\mathcal{N}_i$ as the set of neighbors of node v_i. Aggregation within the tangent space is then conducted as follows:

$$v_i^{l+1,\mathcal{T}} = v_i^{l,\mathcal{T}} + \sum_{j \in \mathcal{N}_i} \frac{1}{|\mathcal{N}_i|} v_j^{l,\mathcal{T}}. \tag{7}$$

Here, the degree-normalized averaging ensures stable gradient flow across layers and prevents the embedding scale from growing disproportionately with layer depth. To further mitigate over-smoothing and vanishing gradient is-

sues, we incorporate skip connections across layers. Instead of relying solely on the top layer, we accumulate the outputs from each intermediate layer: $\mathbf{v}^{sum,\mathcal{T}} = \sum_l(\mathbf{v}^{1,\mathcal{T}}, \mathbf{v}^{2,\mathcal{T}}, ..., \mathbf{v}^{l,\mathcal{T}})$. Finally, this aggregated representation is projected back into the hyperbolic space via the exp map:

$$\mathrm{R}^H = \exp_{\mathbf{o}}^K(\mathbf{p}^{sum,\mathcal{T}}). \tag{8}$$

VGAE-driven Representation Enhancement. To further enhance the expressiveness of node embeddings, we introduce a variational graph autoencoder (VGAE) module that transforms deterministic node features into distributions over latent variables. The encoder consists of a two-stage architecture. The first layer applies a non-linear graph convolution to the hyperbolic output R^H: $R^{(1)} = f_{\mathrm{ReLU}}\left(\mathrm{R}^H, A \mid W^{(0)}\right)$; $R_\mu^{(2)} = f_{\text{Linear}}\left(R^{(1)}, A \mid W_\mu^{(1)}\right)$, $R_\sigma^{(2)} = f_{\text{Linear}}\left(R^{(1)}, A \mid W_\sigma^{(1)}\right)$.

The **inference model** assumes a factorized Gaussian posterior over the latent space. For each node r_i, the approximate posterior distribution is modeled as:

$$\begin{aligned} q(R \mid \mathrm{R}^H, A) &= \prod_{i=1}^{N} q(r_i \mid \mathrm{R}^H, A), \\ q(r_i \mid \mathrm{R}^H, A) &= \mathcal{N}(r_i \mid \mu_{r_i}, \mathrm{diag}(\sigma_{r_i}^2)), \end{aligned} \tag{9}$$

where $\mu_{r_i} = R_\mu^{(2)}[i,:]$ and $\sigma_{r_i}^2 = R_\sigma^{(2)}[i,:]$ are the mean and variance vectors of node r_i, respectively. The potential representation z_i can be computed using mean and variance:$r_i = \mu_i + \sigma \odot \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0,1)$ denotes a standard normal noise vector, and $\odot$ refers to the element-wise (Hadamard) product.

For the **generative model** of the VGAE framework, we adopt an inner-product decoder to reconstruct the observed adjacency structure from the latent representations. Specifically, the conditional likelihood of the adjacency matrix A is modeled as:

$$\begin{aligned} p(A \mid R) &= \prod_{i=1}^{N}\prod_{j=1}^{N} p(a_{ij} \mid r_i, r_j), \\ \text{where } p(a_{ij} = 1 \mid r_i, r_j) &= \sigma(r_i^\top r_j), \end{aligned} \tag{10}$$

where a_{ij} denotes the binary edge indicator between nodes r_i and r_j, and $\sigma(\cdot)$ is the sigmoid activation function. This formulation assumes edge independence and uses latent variable similarity to model edge probability. To regularize the posterior and promote structured latent representations, we incorporate a Kullback-Leibler (KL) divergence penalty between the approximate posterior $q(R \mid \mathrm{R}^H, A)$ and the standard normal prior $p(R)$. The KL loss term is defined as:

$$\begin{aligned} \mathcal{L}_{KL} &= \mathrm{KL}[q(R \mid \mathrm{R}^H, A) \| p(R)] \\ &= \frac{1}{2}\sum_{i=1}^{N}\left(1 + \log(\sigma_{r_i}^2) - \mu_{r_i}^2 - \sigma_{r_i}^2\right), \end{aligned} \tag{11}$$

where μ_{r_i} and $\sigma_{r_i}^2$ are the estimated mean and variance of node r_i's posterior distribution. The reconstruction objective consists of binary cross-entropy terms over positive and negative samples:

$$\begin{aligned} \mathcal{L}_{\text{reco}} &= \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg}} \\ &= -\sum_{(i,j)\in \text{Pos}} \log(\sigma(r_i \cdot r_j)) - \sum_{(i,j)\in \text{Neg}} \log(1 - \sigma(r_i \cdot r_j)). \end{aligned} \tag{12}$$

Bringing together the generative and inference objectives, we define the variational lower bound of the log-likelihood as the final training objective:

$$\begin{aligned} \tilde{\mathcal{L}}_{\text{reco}} &= \mathcal{L}_{\text{pos}} + \mathcal{L}_{\text{neg}} - \mathcal{L}_{KL} \\ &= \sum_{i,j=1}^{N} \mathbb{E}_{r_i, r_j \sim q(\cdot \mid \mathrm{R}^H, A)} \left[\log p(a_{ij} \mid r_i, r_j)\right] \\ &\quad - \mathrm{KL}\left(q(r_i \mid \mathrm{R}^H, A) \| p(r_i)\right). \end{aligned} \tag{13}$$

MOE-Augmented RPMamba

After VGAE refinement, we restore item embeddings in hyperbolic space using lightweight GCN layers, reconstructing each representation via neighborhood averaging:

$$e_i^{l+1} = e_i^l + \sum_{j\in\mathcal{N}_i} \frac{1}{|\mathcal{N}_i|} e_j^l, \quad \bar{e}^l = \sum_{l=1}^{L} e_i^l. \tag{14}$$

Here, L is the number of propagation layers, and $\mathcal{N}_i$ denotes the first-order neighbors of item p_i. We apply a threshold b to filter weak edges in the reconstructed graph based on the predicted connection probability. To model user dynamics, we augment check-in sequences with Rotary Positional Encoding (RoPE) (Su et al. 2024), which preserves relative positional dependencies. Given the base positional vector $\mathbf{p}_i$ and its associated rotation matrix $\mathbf{R}(i)$:

$$\mathbf{R}(i) = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix}, \tag{15}$$

we embed each user sequence as: $\mathrm{E}_u = \left[\left(\hat{e}_{s_1^u} + \mathbf{R}(1)\mathbf{p}_1\right), \cdots, \left(\hat{e}_{s_t^u} + \mathbf{R}(t)\mathbf{p}_t\right)\right]$. The resulting sequence E_u is fed into a RPMamba block to capture long-range sequential dependencies. Instead of using a conventional FFN within Mamba, we incorporate a MOE module to improve its expressiveness and dynamic capacity. Specifically, MOE replaces the static FFN with a gated expert routing mechanism. Let $\mathbf{x}_i$ be the i-th token (i.e., RoPE-enhanced item embedding), then the MOE layer is defined as:

$$\mathrm{MOE}(\mathbf{x}_i) = \sum_{k=1}^{K} \alpha_{i,k} \cdot \mathrm{Expert}_k(\mathbf{x}_i), \tag{16}$$

where K is the number of experts, each $\mathrm{Expert}_k(\cdot)$ is an independent lightweight MLP, and $\alpha_{i,k}$ is the routing weight determined by a gating function:

$$\alpha_{i,k} = \frac{\exp\left(g_k(\mathbf{x}_i)\right)}{\sum_{k'=1}^{K} \exp\left(g_{k'}(\mathbf{x}_i)\right)}, \tag{17}$$

with $g_k(\cdot)$ representing the gating network output for expert k. Sparse expert activation and dynamic routing enable efficient computation and adaptive expert selection for

diverse behaviors. After sequence modeling by the MOE-augmented RPMamba, the final output is:

$$\bar{\mathrm{E}}_u = \mathrm{RPMamba}_{\mathrm{MOE}}(\mathrm{E}_u), \tag{18}$$

which encapsulates both sequential order and semantic complexity. To perform next-item prediction, we compute the relevance score between the final user representation and the target item as:

$$\bar{y} = \bar{\mathrm{E}}_{u,t} \cdot \tilde{e}_{s^u_{t+1}}. \tag{19}$$

Model Training We adopt a multi-objective loss function that jointly considers sequence recommendation accuracy, graph reconstruction fidelity, and parameter regularization. For the primary recommendation task, we employ a binary cross-entropy loss that evaluates whether the model can correctly distinguish the next item in the sequence from a negatively sampled item. Additionally, we leverage subsequence augmentation to enhance generalization: for each check-in sequence $S^u = \{s^u_1, s^u_2, \ldots, s^u_t\}$, we consider all prefix subsequences of the form $\{(s^u_1), (s^u_1, s^u_2), \ldots, (s^u_1, \ldots, s^u_{t-1})\}$ during training. The loss for the main prediction task is defined as:

$$\begin{aligned}\mathcal{L}_{\text{main}} = -\sum_{u\in\mathcal{U}}\sum_{1\le t\le l_u} &\log\sigma\left(\bar{\mathrm{E}}_{u,t}\cdot\tilde{e}_{s^u_{t+1}}\right)\\ &+\log\left(1-\sigma\left(\bar{\mathrm{E}}_{u,t}\cdot\tilde{e}_{p^-_t}\right)\right),\end{aligned} \tag{20}$$

where $\bar{\mathrm{E}}_{u,t}$ is the user embedding derived from the augmented RPMamba model up to time t, and $p^-_t \notin S^u$ is the t-th item randomly chosen from the negative samples. To avoid overfitting, we introduce an L_2 regularization term over model parameters, covering the encoder, decoder, and recommendation components: $\mathcal{L}_{\text{reg}} = \|\theta_{\text{en}}\|_2^2 + \|\theta_{\text{de}}\|_2^2 + \|\theta_{\text{recom}}\|_2^2$. Furthermore, we include the VGAE-based graph reconstruction loss $\tilde{\mathcal{L}}_{\text{reco}}$ defined in the previous section, which encourages the learned latent representations to preserve item-level structural relations. The final training objective is a weighted sum of the above components:

$$\mathcal{L} = \alpha\tilde{\mathcal{L}}_{\text{reco}} + \beta\mathcal{L}_{\text{main}} + \gamma\mathcal{L}_{\text{reg}}, \tag{21}$$

where α, β, and γ are balancing hyperparameters controlling the relative importance of structure reconstruction, recommendation accuracy, and regularization.

Experiments

We aim to answer the following research questions: **RQ1**: How does HM2Rec compare with comprehensive models on performance? **RQ2**: What are the contributions of the key components in HM2Rec, including hyperbolic feature enhancement, VGAE-driven structure refinement, Rotary Positional Encoding, and the MOE architecture? **RQ3**: How robust is HM2Rec when facing noisy user interaction data? **RQ4**: What are the impacts of the essential parameters?

Experimental Setup

Datasets, Evaluation Metrics and Baselines. To evaluate the performance of our proposed model on the sequential recommendation task, we conduct experiments on four widely used public datasets: Books and Toys collected from Amazon (*https://www.amazon.com/*), NYC and TKY collected from Foursquare (Yang et al. 2013). Following recent works (Ye, Xia, and Huang 2023; Qi et al. 2024), we filter out items/users with fewer than 3 interactions in Books and Toys, and fewer than 5 in NYC and TKY. The basic statistics of the datasets are summarized in Table 1. We adopt two widely-used ranking metrics to evaluate the performance: Hit Rate (HR@K) and Normalized Discounted Cumulative Gain (NDCG@K) with $\mathbf{K} = \{10, 20\}$.

Dataset	#Users	#Items	#Interactions	Ave.len.	Density
Books	93,043	54,756	506,637	5.45	$9.94e^{-5}$
Toys	116,429	54,784	478,460	4.11	$1.12e^{-4}$
NYC	1,083	9,989	179,468	165.71	$1.66e^{-2}$
TKY	2,293	15,177	494,807	215.79	$1.42e^{-2}$

Table 1: Statistics of datasets.

Baselines. To demonstrate the effectiveness of HM2Rec model, we compare it with a comprehensive set of sequential recommendation baselines, including RNN-based Methods: **GRU4Rec** (Tan, Xu, and Liu 2016) and **NARM** (Li et al. 2017); Transformer-based Methods: **SASRec** (Kang and McAuley 2018), **BERT4Rec** (Sun et al. 2019), **TiSASRec** (Li, Wang, and McAuley 2020); GNN-based Methods: **SRGNN** (Wu et al. 2019), **SINE** (Tan et al. 2021), **CORE** (Hou et al. 2022), **FEARec** (Du et al. 2023) and **MAERec** (Ye, Xia, and Huang 2023); Contrastive Learning-based Methods: **CL4SRec** (Xie et al. 2022), **DuoRec** (Qiu et al. 2022), **DiffRec** (Wang et al. 2023); and other strong baselines: **MCLRec** (Qin et al. 2023), **AdaMCT** (Jiang et al. 2023), **Mamba4Rec** (Liu et al. 2024a), **MiaSRec** (Choi et al. 2024) **TALE** (Park et al. 2025).

Implementation Details. Our method is implemented in PyTorch and all experiments are conducted on two NVIDIA RTX 4090 GPUs. We use the Adam optimizer with a learning rate of 1e$-$2 and set the number of GCN layers to 2. The embedding dimensions are set to 32 for Books and Toys, and 128 for NYC and TKY. Batch sizes are 2048 for Books and Toys, and 256 for NYC and TKY. The reconstruction threshold b is set to 0.5 (Books), 0.3 (Toys), and 0.6 (NYC and TKY). The number of experts in the HM2Rec module is 4 for Books, Toys, and NYC, and 5 for TKY. Other hyperparameters are kept consistent across datasets: regularization coefficient 1e$-$6, dropout rate 0.3, n-hop neighborhood distance 3, hyperbolic curvature $K = 1$, state-space expansion factor 1, convolution kernel size 4, and top-k routing in MOE set to 2.

Overall Performance Comparison (RQ1) We evaluate the overall performance of HM2Rec against state-of-the-art baselines on four public datasets, as summarized in Table 2. HM2Rec consistently delivers the best results across all metrics and datasets. Among competing methods, MCLRec performs strongly on Books and NYC due to its model-level collaborative learning, while Mamba4Rec achieves competitive results on Toys and TKY by leveraging a state-space formulation that effectively captures long-range dependencies. However, both exhibit limited generalization: MCLRec

Datasets	Books				Toys				NYC				TKY			
Metric	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20	H@10	H@20	N@10	N@20
GRU4Rec	0.6754	0.7787	0.4940	0.5201	0.5194	0.6498	0.3454	0.3783	0.7913	0.8707	0.6829	0.7029	0.8644	0.9098	0.7447	0.7587
NARM	0.6942	0.7901	0.5164	0.5406	0.5306	0.6569	0.3614	0.3933	0.8098	0.8670	0.7219	0.7363	0.8892	0.9115	0.7931	0.8041
BERT4Rec	0.5824	0.6933	0.4036	0.4316	0.4284	0.5650	0.2702	0.3046	0.7775	0.8476	0.6829	0.7005	0.8561	0.9054	0.7463	0.7614
SASRec	0.6491	0.7452	0.4846	0.5089	0.4575	0.5770	0.3183	0.3484	0.7895	0.8578	0.6902	0.7076	0.8666	0.9167	0.7527	0.7653
TiSASRec	0.7063	0.7958	0.5404	0.5630	0.5402	0.6594	0.3817	0.4117	0.8236	0.8601	0.7304	0.7454	0.8892	0.9185	0.7955	0.8054
SRGNN	0.6361	0.7400	0.4662	0.4924	0.4744	0.6055	0.3072	0.3402	0.8006	0.8587	0.7058	0.7204	0.8783	0.9189	0.7805	0.7907
SINE	0.6863	0.7886	0.4929	0.5188	0.5369	0.6590	0.3604	0.3912	0.7442	0.8153	0.6416	0.6595	0.8382	0.8936	0.7184	0.7322
CORE	0.7099	0.7963	0.5421	0.5639	0.5356	0.6527	0.3807	0.4102	0.8172	0.8717	0.7142	0.7280	0.8901	0.9179	0.7850	0.7948
FEARec	0.7074	0.7950	0.5381	0.5603	0.5505	0.6701	0.3916	0.4218	0.8135	0.8763	0.7235	0.7395	0.8945	0.9115	0.7949	0.8042
MAERec	0.7527	0.8348	0.5502	0.5736	0.5932	0.6900	0.4052	0.4346	0.7830	0.8421	0.6300	0.6447	0.8173	0.8853	0.6516	0.6688
CL4SRec	0.6713	0.7743	0.4882	0.5141	0.5110	0.6338	0.3464	0.3773	0.6704	0.7673	0.5378	0.5623	0.8125	0.8766	0.6693	0.6855
DuoRec	0.7637	0.8328	0.5710	0.5943	0.5157	0.6265	0.3544	0.3822	0.8061	0.8726	0.7139	0.7306	0.9001	0.9211	0.7994	0.8021
DiffRec	0.3964	0.4830	0.2917	0.3135	0.2950	0.3896	0.1968	0.2205	0.8061	0.8476	0.7198	0.7302	0.8513	0.8932	0.7408	0.7514
MCLRec	0.7640	0.8351	0.5653	0.5909	0.5149	0.6304	0.3564	0.3855	0.8319	0.8768	0.7457	0.7455	0.8823	0.9206	0.7832	0.7930
AdaMCT	0.7036	0.7962	0.5350	0.5584	0.5330	0.6548	0.3771	0.4078	0.8144	0.8717	0.7199	0.7364	0.8905	0.9111	0.7935	0.8003
Mamba4Rec	0.7261	0.8100	0.5662	0.5824	0.5726	0.6876	0.4180	0.4395	0.8042	0.8680	0.7104	0.7264	0.9006	0.9220	0.7995	0.8057
MiaSRec	0.4985	0.6130	0.3566	0.3854	0.3309	0.4535	0.2196	0.2503	0.6422	0.6881	0.5589	0.5700	0.7087	0.7826	0.6469	0.6651
TALE	0.5218	0.5443	0.4299	0.4356	0.3475	0.3957	0.2618	0.2739	0.4404	0.4958	0.3530	0.3672	0.4575	0.4967	0.3651	0.3750
HM2Rec	**0.7874**	**0.8487**	**0.6053**	**0.6209**	**0.6295**	**0.7262**	**0.4383**	**0.4629**	**0.8458**	**0.8984**	**0.7663**	**0.7781**	**0.9036**	**0.9368**	**0.8269**	**0.8353**
Improv.	3.06%	1.63%	6.01%	4.48%	6.12%	5.25%	4.86%	5.32%	1.67%	2.46%	2.76%	4.37%	0.33%	1.60%	3.43%	3.67%

Table 2: Overall performance. The best performing baseline and best performer in each row are underlined and boldfaced, respectively.

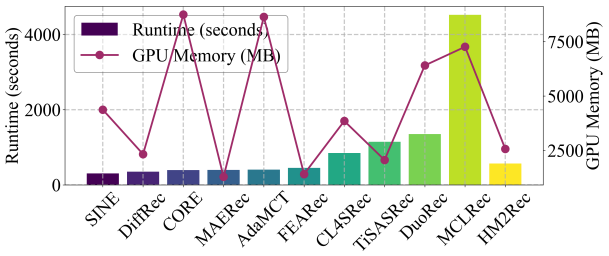

Figure 2: Runtime and GPU memory comparison on NYC.

Dataset	Books		NYC	
Variants	HR	NDCG	HR	NDCG
w/o Hyperbolic	0.8162	0.6022	0.8857	0.7473
w/o VGAE	0.8047	0.5936	0.8809	0.7113
w/o RoPE	0.8125	0.5982	0.8747	0.7389
w/o MOE	0.8335	0.6015	0.8910	0.7584
HM2Rec	**0.8487**	**0.6209**	**0.8984**	**0.7781**

Table 3: Ablation study on Books and NYC datasets.

lacks structural enhancement in representation learning, and Mamba4Rec neglects graph-based prior knowledge and depends on static FFNs. In contrast, HM2Rec jointly integrates hyperbolic graph convolution and VGAE-based structure refinement to capture hierarchical item relations and enhance robustness. The incorporation of RoPE and MOE modules further enables adaptive sequential modeling and precise temporal encoding. Consequently, HM2Rec maintains stable and superior performance across all datasets and evaluation metrics.

Cost Comparison We further compare the runtime and GPU memory of HM2Rec with ten representative baselines, as shown in Figure 2. Despite integrating multiple modules, HM2Rec maintains moderate computational overhead.

Ablation Study (RQ2) To assess the contribution of each component in HM2Rec, we conduct ablation studies on the Books and NYC datasets, and report HR@20 and NDCG@20 in Table 3. The following variants are evaluated: *w/o Hyperbolic*: Replace hyperbolic GCN with a standard Euclidean GCN. *w/o VGAE*: Remove the VGAE and use raw graph embeddings. *w/o RoPE*: Remove the RoPE from the Mamba sequence encoder. *w/o MOE*: Replace the MOE layer in Mamba with a standard FFN. As shown in Table 3, removing any component causes a clear performance drop, indicating that hyperbolic representation, VGAE-based refinement, RoPE, and MOE all contribute to model effectiveness. The complete HM2Rec yields the best results, validating its overall design.

Model Robustness Test (RQ3) We evaluate the robustness of our proposed model, HM2Rec, under varying levels of noise in user interaction data. Specifically, we simulate noisy environments by replacing 5%, 15%, and 25% of the original interactions with randomly sampled negative items. We then compare the performance of HM2Rec against strong baselines, including MCLRec and Mamba4Rec. As shown in Figure 3, HM2Rec consistently achieves superior results across all noise settings on two datasets.

Hyper-parameters Analysis (RQ4)

Embedding Dimension We conduct a hyperparameter study on embedding dimensions using values from {32, 64, 128, 256, 512}. As shown in Figure 4, our results indicate

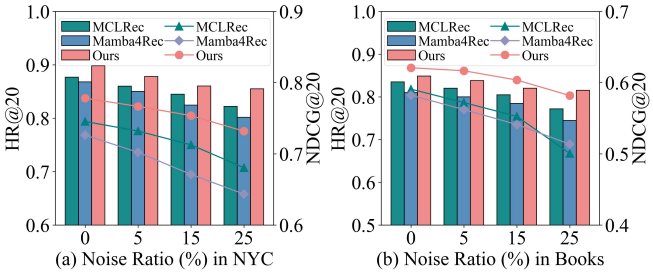

Figure 3: Performance w.r.t. noise ratio on NYC and Books.

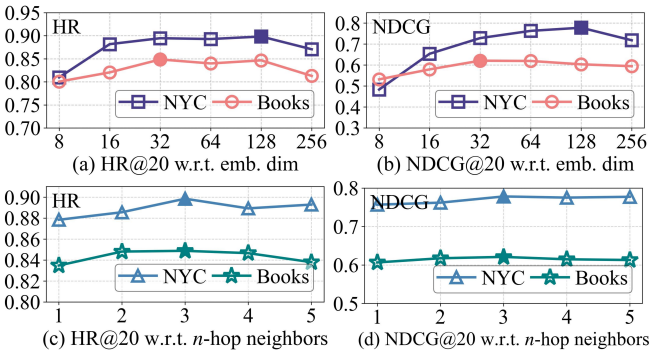

Figure 4: Performance vs. emb. dim. and n-hop neighbors.

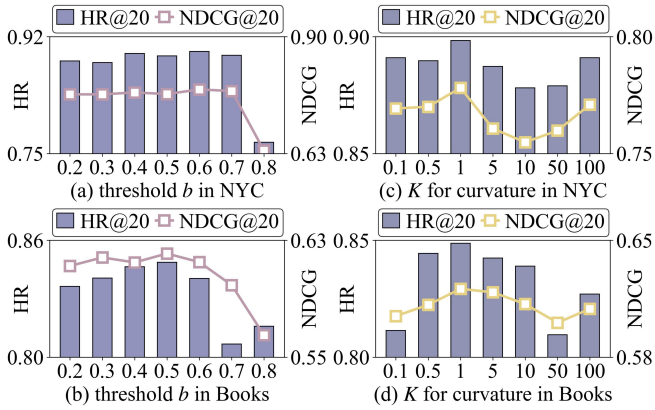

Figure 5: Performance w.r.t. threshold b and curvature K.

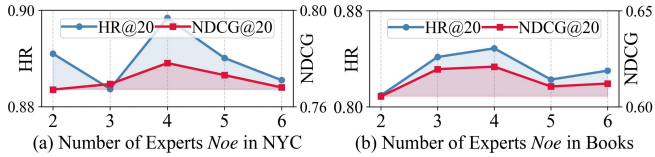

Figure 6: Performance w.r.t. different number of experts.

that smaller dimensions are optimal for Books and Toys datasets with 32 achieving top performance. Conversely, the best results on NYC and TKY datasets were obtained with larger dimensions (128).

Effect of n-hop Neighbors. We further investigate the impact of the n-hop neighborhood size used in constructing the user-item interaction graph. We test values of $n \in \{1, 2, 3, 4, 5\}$ on two representative datasets: Books and NYC. As shown in Figure 4, performance improves as n increases initially, peaks at $n = 3$, and then begins to decline. This shows that a 3-hop neighborhood strikes a good balance between capturing sufficient structural context and avoiding excessive noise from distant, less relevant nodes. Finally, we select $n = 3$ as the default setting for all datasets.

Effect of Reconstruction Threshold b. During graph reconstruction, we introduce a threshold parameter b to determine whether an edge should be retained. We evaluate candidate values $b \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$ on the Books and NYC datasets. As shown in Figure 5, model performance is sensitive to the choice of b. Lower thresholds tend to introduce noisy edges, while higher thresholds may omit important connections. Empirically, we find that $b = 0.5$ works best for Books, 0.3 for Toys, and 0.6 for NYC and TKY. These values are used in all corresponding experiments.

Effect of Hyperbolic Curvature K. We analyze the impact of the curvature parameter K used in the hyperboloid model, which controls the degree of negative curvature in the embedding space. We test $K \in \{0.1, 0.5, 1, 5, 10, 50, 100\}$ on the Books and NYC datasets. As shown in Figure 5, performance is sensitive to extreme curvature values. Too small or too large K may distort the geometry and harm representation quality. We observe that $K = 1$ consistently yields the best results across all datasets, and thus adopt it as the default setting.

Effect of Number of Experts. We study the effect of the number of experts in HM2Rec by testing $\{2, 3, 4, 5, 6\}$ on the Books and NYC datasets. As shown in Figure 6, performance improves with more experts but plateaus or slightly declines beyond a certain point, indicating that a moderate number balances capacity and generalization. Accordingly, we set the number of experts to 4 for Books, Toys, and NYC, and 5 for TKY in all experiments.

Conclusion

In this work, we proposed HM2Rec, a unified framework for sequential recommendation that integrates hyperbolic representation learning, structure-aware graph reconstruction, and dynamic sequence modeling. Specifically, HM2Rec addresses two key challenges in sequential recommendation: capturing hierarchical user-item relationships and enhancing the adaptability of sequence encoders. By combining hyperbolic GCNs, VGAE-based structure refinement, and a MOE-Augmented RPMamba, our model learns robust and expressive representations. Extensive experiments on four public datasets demonstrate that HM2Rec consistently outperforms state-of-the-art methods, validating the effectiveness of each component and the overall framework.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 62572486, No. 62302211), the Natural Science Foundation of Shandong Province (No. ZR2023MF007), and the State Key Laboratory of Novel Software Technology (KFKT2024A03), and was also funded by the European Union under the Horizon Europe programme (grant numbers: 101135930, 101168438), in part by the UKRI through the UK Government's Horizon Europe funding Guarantee (grant numbers: 10119564, 10126241).

References

Bach, N. X.; Long, D. H.; and Phuong, T. M. 2020. Recurrent convolutional networks for session-based recommendations. *Neurocomputing*, 411: 247–258.

Chami, I.; Ying, Z.; Ré, C.; and Leskovec, J. 2019. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32.

Chen, H.; Zhou, K.; Jiang, Z.; Yeh, C.-C. M.; Li, X.; Pan, M.; Zheng, Y.; Hu, X.; and Yang, H. 2023. Probabilistic Masked Attention Networks for Explainable Sequential Recommendation. In *IJCAI*, 2068–2076.

Chen, J.; Wang, H.; and Shang, J. 2024. GeoMamba: Towards Efficient Geography-aware Sequential POI Recommendation. *IEEE Access*.

Choi, M.; Kim, H.-y.; Cho, H.; and Lee, J. 2024. Multi-intent-aware session-based recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, 2532–2536.

Donkers, T.; Loepp, B.; and Ziegler, J. 2017. Sequential user-based recurrent neural network recommendations. In *Proceedings of the eleventh ACM conference on recommender systems*, 152–160.

Du, X.; Yuan, H.; Zhao, P.; Qu, J.; Zhuang, F.; Liu, G.; Liu, Y.; and Sheng, V. S. 2023. Frequency enhanced hybrid attention network for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 78–88.

Fan, H.; Zhu, M.; Hu, Y.; Feng, H.; He, Z.; Liu, H.; and Liu, Q. 2024. TiM4Rec: An Efficient Sequential Recommendation Model Based on Time-Aware Structured State Space Duality Model. *arXiv preprint arXiv:2409.16182*.

Fan, Z.; Liu, Z.; Wang, Y.; Wang, A.; Nazari, Z.; Zheng, L.; Peng, H.; and Yu, P. S. 2022. Sequential recommendation via stochastic self-attention. In *Proceedings of the ACM web conference 2022*, 2036–2047.

Gu, A.; and Dao, T. 2024. Mamba: Linear-Time Sequence Modeling with Selective State Spaces.

He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 639–648.

Hou, Y.; Hu, B.; Zhang, Z.; and Zhao, W. X. 2022. Core: simple and effective session-based recommendation within consistent representation space. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 1796–1801.

Jiang, J.; Zhang, P.; Luo, Y.; Li, C.; Kim, J. B.; Zhang, K.; Wang, S.; Xie, X.; and Kim, S. 2023. AdaMCT: adaptive mixture of CNN-transformer for sequential recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 976–986.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Li, J.; Ren, P.; Chen, Z.; Ren, Z.; Lian, T.; and Ma, J. 2017. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 1419–1428.

Li, J.; Wang, Y.; and McAuley, J. 2020. Time interval aware self-attention for sequential recommendation. In *Proceedings of the 13th international conference on web search and data mining*, 322–330.

Li, Y.; Chen, H.; Sun, X.; Sun, Z.; Li, L.; Cui, L.; Yu, P. S.; and Xu, G. 2021. Hyperbolic hypergraphs for sequential recommendation. In *Proceedings of the 30th ACM international conference on information & knowledge management*, 988–997.

Liao, X.; Liu, W.; Chen, C.; Zhou, P.; Zhu, H.; Tan, Y.; Wang, J.; and Qi, Y. 2023. HyperFed: hyperbolic prototypes exploration with consistent aggregation for non-IID data in federated learning. *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligenc*, 3957–3965.

Liu, C.; Lin, J.; Wang, J.; Liu, H.; and Caverlee, J. 2024a. Mamba4rec: Towards efficient sequential recommendation with selective state space models. *arXiv preprint arXiv:2403.03900*.

Liu, Y.; Qi, L.; Mao, X.; Liu, W.; Wang, F.; Xu, X.; Zhang, X.; Dou, W.; Zhou, X.; and Beheshti, A. 2025a. Hyperbolic Variational Graph Auto-Encoder for Next POI Recommendation. In *Proceedings of the ACM on Web Conference 2025*, 3267–3275.

Liu, Y.; Walder, C.; Xie, L.; and Liu, Y. 2024b. Probabilistic Attention for Sequential Recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 1956–1967.

Liu, Y.; Xia, L.; and Huang, C. 2024. Selfgnn: Self-supervised graph neural networks for sequential recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1609–1618.

Liu, Z.; Liu, Q.; Wang, Y.; Wang, W.; Jia, P.; Wang, M.; Liu, Z.; Chang, Y.; and Zhao, X. 2025b. SIGMA: Selective Gated Mamba for Sequential Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 12264–12272.

Ma, C.; Ma, L.; Zhang, Y.; Sun, J.; Liu, X.; and Coates, M. 2020. Memory augmented graph neural networks for sequential recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 5045–5052.

Nickel, M.; and Kiela, D. 2017. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30.

Park, S.; Yoon, M.; Choi, M.; and Lee, J. 2025. Temporal Linear Item-Item Model for Sequential Recommendation. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, 354–362.

Peintner, A. 2023. Sequential Recommendation Models: A Graph-based Perspective. In *Proceedings of the 17th ACM Conference on Recommender Systems*, 1295–1299.

Qi, L.; Liu, Y.; Liu, W.; Pei, S.; Xu, X.; Zhang, X.; Wang, Y.; and Dou, W. 2024. Counterfactual user sequence synthesis augmented with continuous time dynamic preference modeling for sequential POI recommendation. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, 2306–2314.

Qin, X.; Yuan, H.; Zhao, P.; Fang, J.; Zhuang, F.; Liu, G.; Liu, Y.; and Sheng, V. 2023. Meta-optimized contrastive learning for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 89–98.

Qiu, R.; Huang, Z.; Yin, H.; and Wang, Z. 2022. Contrastive learning for representation degeneration problem in sequential recommendation. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, 813–823.

Qu, H.; Zhang, Y.; Ning, L.; Fan, W.; and Li, Q. 2024. Ssd4rec: a structured state space duality model for efficient sequential recommendation. *arXiv preprint arXiv:2409.01192*.

Rajput, S.; Mehta, N.; Singh, A.; Hulikal Keshavan, R.; Vu, T.; Heldt, L.; Hong, L.; Tay, Y.; Tran, V.; Samost, J.; et al. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36: 10299–10315.

Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; and Liu, Y. 2024. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568: 127063.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 1441–1450.

Sun, X.; Chen, L.; and Yang, J. 2019. Learning from web data using adversarial discriminative neural networks for fine-grained classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 273–280.

Sun, X.; Gazagnadou, N.; Sharma, V.; Lyu, L.; Li, H.; and Zheng, L. 2023. Privacy assessment on reconstructed images: Are existing evaluation metrics faithful to human perception? volume 36, 10223–10237.

Tan, Q.; Zhang, J.; Yao, J.; Liu, N.; Zhou, J.; Yang, H.; and Hu, X. 2021. Sparse-interest network for sequential recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*, 598–606.

Tan, Y. K.; Xu, X.; and Liu, Y. 2016. Improved recurrent neural networks for session-based recommendations. In *Proceedings of the 1st workshop on deep learning for recommender systems*, 17–22.

Wang, F.; Chen, C.; Liu, W.; Lei, M.; Chen, J.; Liu, Y.; Zheng, X.; and Yin, J. 2025. DR-VAE: Debiased and Representation-enhanced Variational Autoencoder for Collaborative Recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 12703–12711.

Wang, W.; Xu, Y.; Feng, F.; Lin, X.; He, X.; and Chua, T.-S. 2023. Diffusion recommender model. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 832–841.

Wu, S.; Tang, Y.; Zhu, Y.; Wang, L.; Xie, X.; and Tan, T. 2019. Session-based recommendation with graph neural networks. In *AAAI*, volume 33, 346–353.

Xie, X.; Sun, F.; Liu, Z.; Wu, S.; Gao, J.; Zhang, J.; Ding, B.; and Cui, B. 2022. Contrastive learning for sequential recommendation. In *2022 IEEE 38th international conference on data engineering (ICDE)*, 1259–1273. IEEE.

Yang, D.; Zhang, D.; Yu, Z.; and Wang, Z. 2013. A sentiment-enhanced personalized location recommendation system. In *Proceedings of the 24th ACM conference on hypertext and social media*, 119–128.

Yang, Y.; Huang, C.; Xia, L.; Liang, Y.; Yu, Y.; and Li, C. 2022. Multi-behavior hypergraph-enhanced transformer for sequential recommendation. In *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, 2263–2274.

Yang, Y.; Wu, L.; Zhang, K.; Hong, R.; Zhou, H.; Zhang, Z.; Zhou, J.; and Wang, M. 2023. Hyperbolic graph learning for social recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 36(12): 8488–8501.

Ye, Y.; Xia, L.; and Huang, C. 2023. Graph masked autoencoder for sequential recommendation. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 321–330.

Yuan, M.; Xiao, Y.; Chen, W.; Zhao, C.; Wang, D.; and Zhuang, F. 2025. Hyperbolic Diffusion Recommender Model. In *Proceedings of the ACM on Web Conference 2025*, 1992–2006.

Zhang, H.; Wang, H.; Wang, G.; Liu, J.; and Liu, Q. 2022. A hyperbolic-to-hyperbolic user representation with multi-aspect for social recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 4667–4671.

Zhang, L.; and Wu, N. 2024. HGCH: A Hyperbolic Graph Convolution Network Model for Heterogeneous Collaborative Graph Recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 3186–3196.