Enhancing Session-Based Recommendation With Multi-Interest Hyperbolic Representation Networks

Tongcun Liu, Xukai Bao, Jiaxin Zhang, Kai Fang, *Member, IEEE*, and Hailin Feng, *Member, IEEE*

***Abstract*— Session-based recommendation (SBR) aims to predict the next item a user might click within an ongoing session, without relying on user profiles or historical data. Modern approaches typically use graph networks to learn item embeddings in Euclidean space via graph convolution operations. However, they often struggle to capture the diversity of user interactions within short, hierarchically structured sessions, which is essential for accurate predictions in SBR. To tackle these challenges, we propose a multi-interest hyperbolic representation network (MIHRN) to enhance the performance of SBR by adeptly modeling both intricate high-order spatial structures and sequence relationships among items in hyperbolic geometry space. Specifically, we use a hyperbolic hypergraph neural network to exploit the high-order spatial relationships and local clustering structures inherent within sessions. Subsequently, a multiaspect interest representation module is designed to articulate the diversity of user interests. Extensive experiments on three real-world datasets demonstrate that the proposed method achieves performance improvements of 23.81%, 14.81%, and 36.84%, respectively, under the P@10 metric.**

***Index Terms*— Hyperbolic representation, hypergraph convolution, multiaspect interest, session-based recommendation (SBR).**

NOMENCLATURE

Dataset	*Explanation*
$\mathcal{V}, \mathcal{S}$	Set of items and sessions.
d	Embedding dimension.
c	Negative curvature in Lorentz model.
$\mathbf{o}$	Reference point to perform mapping transformations.
T	Session length.
$\mathcal{G}$	Undirect hypergraph.
$\mathbf{I}$	Incidence matrix in hypergraph.
$\Gamma_{\mathbf{x}}^{\exp,c}(\cdot)$	Exponential map function with reference point $\mathbf{x}$.
$\Gamma_{\mathbf{x}}^{\log,c}(\cdot)$	Logarithmic map function with reference point $\mathbf{x}$.
$\mathbf{P}^{\mathbb{E},0}$	Initialized Euclidean embedding of all items in $\mathcal{V}$.
$\mathbf{P}^{\mathbb{H},0}$	Initialized hyperbolic embedding of all items in $\mathcal{V}$.
$\mathbf{P}^{\mathbb{H},L}$	Hyperbolic embedding after L layer convolution.
$\mathbf{E}^{\mathbb{E},p}$	Reversed position embedding matrix.
$\mathbf{E}^{\mathbb{H}}$	Hyperbolic embedding matrix of given session.
$\mathbf{C}^{(m)}$	Affinity matrix of mth aspect interest.
$\mathbf{U}$	Multiple interest matrix.
$\mathbf{H}$	Weighted hidden hyperbolic representation for session s.
$\widetilde{\mathbf{X}}_{1:T}$	Item representation in the feedforward network.
θ	Weighted session representation.

Received 24 February 2024; revised 3 July 2024 and 15 September 2024; accepted 17 November 2024. This work was supported in part by the Natural Science Foundation of Zhejiang Province under Grant LGG22F020010 and in part by the Research and Development Fund Talent Start-Up Project of Zhejiang A&F University under Grant 2019FR070. *(Corresponding authors: Hailin Feng; Kai Fang.)*

Tongcun Liu, Kai Fang, and Hailin Feng are with the College of Mathematics and Computer Science, Zhejiang A&F University, Hangzhou 311300, China (e-mail: tongcun.liu@gmail.com; Kaifang@ieee.org; hlfeng@zafu.edu.cn).

Xukai Bao and Jiaxin Zhang are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: bupt.bxk@gmail.com; zhang_jx@bupt.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2024.3502769

I. INTRODUCTION

RECOMMENDER systems play a vital role in e-commerce and multimedia platforms by effectively advertising relevant information to potential users who are interested in it. Traditional approaches typically concentrate on exploring user preferences through historical user–item interaction data collected from mobile terminals [1]. However, as increasingly stringent privacy policies are implemented, obtaining comprehensive interaction data is becoming increasingly challenging, particularly for anonymous users. A session is defined as a series of activities by a user within a given time period, recorded in chronological order. In such cases, session-based recommendation (SBR), which aims to predict the next potentially clicked item within the ongoing session without using any user profile or historical data [2], [3], [4], has garnered growing attention in both academia and industry.

Intuitively, user's behavior usually exhibits intricate temporal correlation, and great efforts have been devoted to improving the SBR accuracy. Earlier approaches used Markov chain [5], [6] and recurrent neural network (RNN) [7], [8], [9] for SBR, focusing on modeling the sequential transfer relationships between adjacent items in the session but overlooking the global coherence relationship among items. In real-world scenarios, item transitions are frequently influenced by the collective effects of previous item clicks, and many-to-many and

higher order relationships exist among items [10], [11], [12], [13]. For this issue, some studies resort to constructing various graphs (i.e., session graph [2], [7] and global graph [12]) network and incorporate attention mechanism [14], [15], [16] and graph neural networks (GNNs) [3], [11], [12] to capture local and global-level item transition pattern. The intricate neural network architectures used in these methodologies significantly enhance the capacity to learn session representations by exploiting sequence dependencies and low-level spatial structure (i.e., first-order and second-order interaction), leading to improvements in SBR performance. However, traditional graph modeling often neglects higher level interactions among users and items. In contrast, hypergraph, where an edge can connect more than two nodes, can capture higher order dynamic relationships due to their nonpairwise data structure. Consequently, hypergraph-based modeling approaches for SBR have been developed [17], [18], [19], [20].

While these graph-based methods have made notable advancements, they solely optimize the overall model based on the final prediction outcomes, which makes it challenging to address the data sparsity issue in anonymous session recommendation and increase the risk of overfitting. The sparsity issue is becoming increasingly severe in hypergraph-based SBR models [21]. More importantly, two inherent attributes of representation distributions embedded within item transitions were also disregarded, yet they are crucial for modeling user preferences in SBR [22]. First, user's preferences usually exhibit hierarchical nature, where item popularity follows a power-law distribution. This implies that a small number of items have high occurrence probabilities, while the majority of items have very few interactions. Corresponding, session length also follows a power-law distribution. As illustrated in Fig. 1(a) and (b), approximately 73% of items were interacted with less than ten times, and roughly 83% of sessions have a length of less than 10. The power-law distribution arises from a tree-like hierarchical organization, indicating that a small group of nodes is hierarchically organized into increasingly larger groups [23], [24]. Attempting to model the hierarchical graph structure in the Euclidean space results in considerable distortion in embedding representation due to the quadratic growth of the volume inside a Euclidean ball with a radius r [25]. Second, a user's interest in a session is typically diverse, and not all the items in the session are relevant to the user's current preference due to accidental or mistake click. Therefore, considering a fixed single embedding schedule (such as last clicked item [8], [26], the average item within the session [17], and a special fixed target item [27]) shared by all the sessions as the user's current preference may lead to model underfitting and inaccurate prediction results. Thus, effectively modeling users' diverse preferences from short, hierarchically structured sessions is a challenging and vital task for SBR.

Recent analyses [22], [28] emphasize that the Euclidean space expands polynomially, which can lead to significant distortion in embedding when modeling sparse hierarchical relational data, thereby compromising the preservation of distances between users and items. In contrast, extensive literature highlights the effectiveness of hyperbolic space in representing data characterized by hierarchical structures [25], [29]. One of its primary advantages lies in its faster expansion rate compared with the Euclidean space, a characteristic derived from hyperbolic geometry. This property endows the hyperbolic space with a more powerful, exponential-level representation capability [25], [30], driving the exploration of representation learning techniques tailored to capture hierarchical behaviors in user interactions.

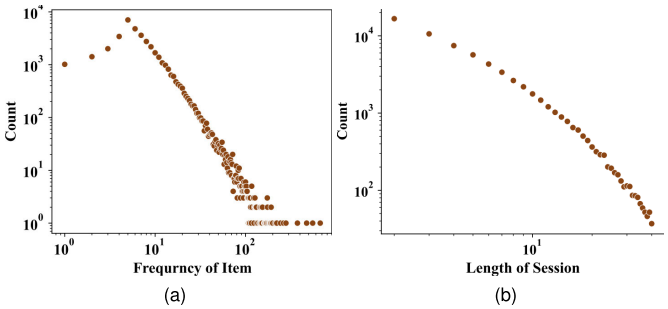


Fig. 1. Distributions of item and session in the Tmall dataset. (a) Distribution of item interaction. (b) Distributions of session length.

To effectively tackle the aforementioned challenge, we propose the multi-interest hyperbolic representation networks (MIHRN) to bolster the performance of SBR systems by adeptly modeling both intricate high-order spatial structures and sequence relationships among items in hyperbolic geometry space. Specifically, we extract hyperbolic semantic features using a hyperbolic hypergraph neural network, which effectively exploits the high-order spatial relationships and local clustering structures inherent within session data. Unlike previous works that focus on chronologically modeling each session, our approach introduces hyperedges to model sessions and uses hyperbolic hypergraph convolution to explore higher order user–item interactions, effectively leveraging the hyperbolic space to accommodate networks with power-law distributions and sparse hierarchical structures. Subsequently, we design a multiaspect interest representation module that uses an attention mechanism to articulate the diversity of a user's interests in the hyperbolic space, thereby generating a weighted hyperbolic session representation. This representation considers the user's current diverse interest features and discerns the importance of each behavior. Compared with prior works that either focus on the last few consecutive items to represent diverse user intents [31], [32] or identify multiple user interests over an extended user–item history using a fixed number of user interests [33], [34], we generate the user's multiple interest matrix by designing learnable affinity matrices in the hyperbolic space. Moreover, to maximize the utility of limited interaction data, we introduce a minimal data augmentation strategy aimed at enhancing the session representation, thereby enhancing learning efficiency.

Our main contributions could be summarized as follows.

1) A hyperbolic hypergraph was designed to model the high-order spatial relationships and local clustering structures inherent within session data. This approach can significantly enhance the quality of item representations by alleviating the effects of the skewed power-law distribution.

2) We developed a multiaspect interest representation module that uses learnable affinity matrices in the hyperbolic space to capture the diversity of a user's interests, seamlessly integrating it into hyperbolic session representation with a weighted schedule.
3) Extensive experiments conducted on three real-world datasets (Tmall, Diginetica, and Nowplaying) demonstrated the superiority of MIHRN over state-of-the-art methods. Compared with the second-based model (HADCG), our MIHRN showed significant improvements in the P@10 metric, with gains of 23.81%, 14.84%, and 36.84%, respectively.

II. Related Work

In this section, we review related work from the SBR to hyperbolic learning in recommendation.

A. SBR

SBR is a specialized subtask within recommendation systems, focusing on analyzing and providing recommendations for anonymous users based solely on their current session activities. Early SBR research relied on traditional methods. Matrix factorization techniques [35], [36] have been widely used in recommendation systems but are not suitable for SBR due to the absence of explicit user information. Item-KNN [37] calculates item similarities based on their co-occurrence within the same session but fails to consider the sequential order of items. Consequently, Markov-chain-based sequential methods emerged to capture order information. For instance, Shani et al. [5] proposed using a Markov decision process (MDP) to compute item transition probabilities, while Rendle et al. [6] introduced FPMC to model sequential behavior between adjacent clicks using a combination of Markov chains and matrix factorization.

In recent years, deep learning models such as RNNs and convolutional neural networks have achieved significant success in natural language processing and have been adopted for SBR tasks. GRU4REC [2], which uses a gated RNN architecture, models the sequential order of clicks in a session. Tan et al. [7] improved upon this with GRU4REC+, incorporating better data augmentation techniques and considering temporal shifts in user behaviors. Li et al. [14] introduced NARM, which combines sequence-to-sequence RNNs with an attention mechanism for SBR. Other notable models include STAMP [15], which captures user interest with a short-term attention/memory priority model, CSRM [38], and CoSAN [39], which enhance models with collaborative neighborhood information. SR-IEM [26] uses an affinity matrix to determine item importance, and DSAN [27] uses dual sparse attention with a fixed target embedding.

Despite the success of RNN-based approaches, they often struggle with learning complex item transitions. To address this, Wu et al. [3] introduced SRGNN, a graph-based model for SBR. Subsequent models have built on this approach: GC-SAN [11] integrated self-attention mechanisms with GNNs, FGNN [40] used a weighted GAT [10] and a readout function to analyze item transitions, and PA-GGAN [41] incorporated reversed position embedding to capture item position information. However, these graph modeling approaches often neglect higher level interactions among user–item interactions. To address this limitation, hypergraph-based models for SBR have been developed, leveraging their ability to capture higher order dynamic relationships. Notable examples include S^2-DHCN [17], DING [18], CoHHN [19], and BiPNet [20]. Despite their advancements, these methods often rely on the last item or fixed approaches to represent the query vector, thus failing to account for the dynamic interests within each session.

To address the diverse preference characteristics of users within a session, several SBR models, such as MSGIFSR [31], Atten-MiXer [32], and TASI-GNN [42], have focused on capturing multiple user intents by analyzing the last few consecutive items. However, these models may struggle to accurately determine user intent when the last item is unimportant or noisy. In addition, some studies have aimed to identify multiple user interests over an extended user–item history by using a fixed number of interests for all users [42], [43]. This approach can miss some user interests or include unnecessary ones, as the number of user interests varies significantly between users. To address this issue, Choi et al. [44] introduced the MiaSRec model, which captures various user intents by generating multiple session representations centered on each item and dynamically selecting the important ones. In contrast, our approach involves developing a multiaspect interest representation module by designing learnable affinity matrices in the hyperbolic space, effectively capturing the diversity of a user's interests.

B. Hyperbolic Learning in Recommendation

Hyperbolic space is a mathematical concept representing a non-Euclidean space. Unlike Euclidean space, where the inner product is positive definite, in the hyperbolic space, the inner product is negative definite. This implies that in hyperbolic space, the angle between two vectors can be greater than 90°, whereas in Euclidean space, the angle is always less than 90°. Recent works [45], [46] have shown that the hyperbolic space offers greater embedding capability, especially in case where graph-structured data display hierarchical and scale-free attributes which is difficult to capture with representations in Euclidean space. Considering graph data usually present a hierarchical structure of power-law distribution, researchers have proposed a series of works that generalize hyperbolic embedding learning to GNNs [28], [47], [48], [49], including HGNN, HGCN, and HGAT. Those works represent nodes in the hyperbolic space and perform graph convolutions by injecting nodes from the hyperbolic space to the tangent space, which build a bridge between GCN and hyperbolic learning.

Noting the potentials of hyperbolic geometry in capturing complex user–item interactions, many researches have leveraged hyperbolic space for recommendation systems. For instance, Sun et al. [50] exploits higher order information in user–item interactions through hyperbolic GCN model; Zhang and Wu [51] improve the existing hyperbolic GCN structure

for collaborative filtering and incorporates side information; Yang et al. [52] design an additional hyperbolic geometric regularization to enhance performance; Wang et al. [53] and Yang et al. [29] explore hierarchy properties of user and item representations with multiple aspects for social recommendation. For SBR, [25] is the first work that exploits hyperbolic item representation for SBR. Afterward, Li et al. [30] propose time-aware hyperbolic graph attention network (TA-HGAT) model, which is a novel hyperbolic GNN framework to build an SBR model considering temporal information, and Su et al. [22] propose a hierarchy-aware dual clustering graph network (HADCG) model to capture hierarchy structures and to enhance hyperbolic representation learning from collaborative perspectives. Unlike previous studies, we introduce hyperedges to represent sessions and explore higher order relationships among user–item interactions within a hypergraph in hyperbolic space. In addition, we capture the diversity of a user's interests by learning multi-interest representations in hyperbolic space using learnable affinity matrices.

III. Methodologies

In this section, we delve into the implementation details of the proposed methodologies. First, we provide an overview of the preliminaries. Subsequently, we offer a detailed exposition of the proposed MIHRN model.

A. Preliminaries

To facilitate the description of the proposed model, this section begins with a concise explanation of key notations and the definition of SBR.

1) Notations: In this article, uppercase bold letters are used to denote matrices (e.g., $\mathbf{Q}$), lowercase bold letters to denote row vectors (e.g., $\mathbf{q}$), and regular typeface letters indicate scalars. Calligraphic letters are used to denote sets (e.g., item set $\mathcal{V}$, session set $\mathcal{S}$), and $|\cdot|$ denotes the size of the set. Nomenclature lists the major notations used in this study.

2) SBR Task: The objective of SBR is to predict the next item a user will click based solely on the user's current session. Let $\mathcal{V} = \{v_1, v_2, v_3, \ldots, v_{|\mathcal{V}|}\}$ denote the unique set of items involved in all the sessions, where $|\mathcal{V}|$ represents the number of items. $\mathcal{S} = \{s_1, s_2, \ldots, s_{|\mathcal{S}|}\}$ represents the set comprising all the historical sessions, where $|\mathcal{S}|$ denotes the total number of sessions. Each session in $\mathcal{S}$ is described by $s = \{v_1^s, v_2^s, v_3^s, \ldots, v_T^s\}$, with the true next clicked item denoted as v_{T+1}^s, where T represents the session length. If the session length exceeds T, only the most recent T interactions are considered; otherwise, s is padded with zeros on the right. Thus, given $\mathcal{V}$, $\mathcal{S}$, and the current session s, the objective of SBR is to build and train a high-accuracy prediction model to generate the top-N ($1 \le N \le |\mathcal{V}|$) items in item set $\mathcal{V}$ with the highest probability of being clicked by the user.

B. Multi-Interest Hyperbolic Representation Networks

The overall workflow of the proposed MIHRN is depicted in Fig. 2, which comprises hypergraph construction, hyperbolic embedding learning, multiaspect interest extractor, interest fusion, and prediction.

1) Hypergraph Construction: The original session data are typically organized as linear sequences, where two items v_i^s and v_{i+1}^s are connected only if a user interacted with item v_i^s before item v_{i+1}^s. To capture the complex high-order relationships in user–item interactions, inspired by [17], we transform the session data into an undirected hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set containing $|\mathcal{V}|$ unique items, and $\mathcal{E}$ is a set of hyperedges containing all the sessions. This transformation allows us to concretize the many-to-many high-order nonlinear relationships. The process of hypergraph construction is illustrated in the left part of Fig. 2. Each hyperedge $\xi \in \mathcal{E}$ represents a session, which contains two or more items. The hypergraph can be represented by an incidence matrix $\mathbf{I} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{E}|}$, where $\mathbf{I}_{i\xi} = 1$ if the hyperedge ξ contains item $v_i \in \mathcal{V}$, otherwise 0. In general, each node in the hypergraph could come with a d-dimensional attribute vector. Therefore, if all the nodes were denoted as $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{V}|}\} \in \mathbb{R}^{|\mathcal{V}|\times d}$, then $\mathcal{G} = (\mathbf{I}, \mathbf{X})$ would be used to represent the entire hypergraph for simplicity.

2) Hyperbolic Embedding Learning: In this section, we first introduce the initialization of item embedding in hyperbolic space, followed by a description of how to learn the semantic embedding of items using hyperbolic hypergraph convolution and hyperbolic session embedding.

a) Embedding initialization: Hyperbolic space is a Riemannian manifold with a constant negative curvature, covered by hyperbolic geometry that increases exponentially rather than polynomially. This property makes it easier to explore hierarchical relationships among items in a given session [22]. Here, the representation from Lorentz hyperbolic space was used for item embedding. Specifically, for each item v_i, an initialized Euclidean embedding $\mathbf{P}_i^{\mathbb{E},0} \in \mathbb{R}^{1\times(d-1)}$ was first sampled from a Gaussian distribution in Euclidean space. We then explicitly transformed the Euclidean embedding to hyperbolic embedding. In the hyperbolic space, we regarded the origin $\mathbf{o} = ((K)^{1/2}, 0, \ldots, 0) \in \mathbb{H}^{d,K}$ as a reference point for performing mapping transformations, where $K = -1/c$ is the reciprocal of the curvature c, a hyperparameter set empirically. Thus, the initialized hyperbolic embedding $\mathbf{P}_i^{\mathbb{H},0}$ can be derived from the initialized Euclidean embedding $\mathbf{P}_i^{\mathbb{E},0}$ via the exponential map function $\Gamma_{\mathbf{o}}^{\exp,c}(\cdot)$, calculated as

$$\mathbf{P}_i^{\mathbb{H},0} = \Gamma_{\mathbf{o}}^{\exp,c}\left(\widetilde{\mathbf{P}}_i^{\mathbb{E},0}\right) = \cosh\left(\frac{\left\|\widetilde{\mathbf{P}}_i^{\mathbb{E},0}\right\|_{\mathcal{L}}}{\sqrt{c}}\right)\mathbf{o} + \sqrt{c}\sinh\left(\frac{\left\|\widetilde{\mathbf{P}}_i^{\mathbb{E},0}\right\|_{\mathcal{L}}}{\sqrt{c}}\right)\frac{\widetilde{\mathbf{P}}_i^{\mathbb{E},0}}{\left\|\widetilde{\mathbf{P}}_i^{\mathbb{E},0}\right\|_{\mathcal{L}}} \quad (1)$$

where $\widetilde{\mathbf{P}}_i^{\mathbb{E},0} = [0; \mathbf{P}_i^{\mathbb{E},0}]$ is a d-dimensional vector in the tangent space, $[;]$ denotes concatenation operation, and $\langle\widetilde{\mathbf{P}}_i^{\mathbb{E},0}, \mathbf{o}\rangle_{\mathcal{L}} = 0$ ensures that $\widetilde{\mathbf{P}}_i^{\mathbb{E},0}$ belongs to the tangent space of the reference point $\mathbf{o}$. Consequently, the initialized hyperbolic embedding of all the items in $\mathcal{V}$ can be represented as matrices $\mathbf{P}^{\mathbb{H},0} \in \mathbb{R}^{|\mathcal{V}|\times d}$. Corresponding, the reverse logarithmic map function $\Gamma_{\mathbf{o}}^{\log,c}(\cdot)$ can map project vectors back to the tangent space centered at the reference point, such that

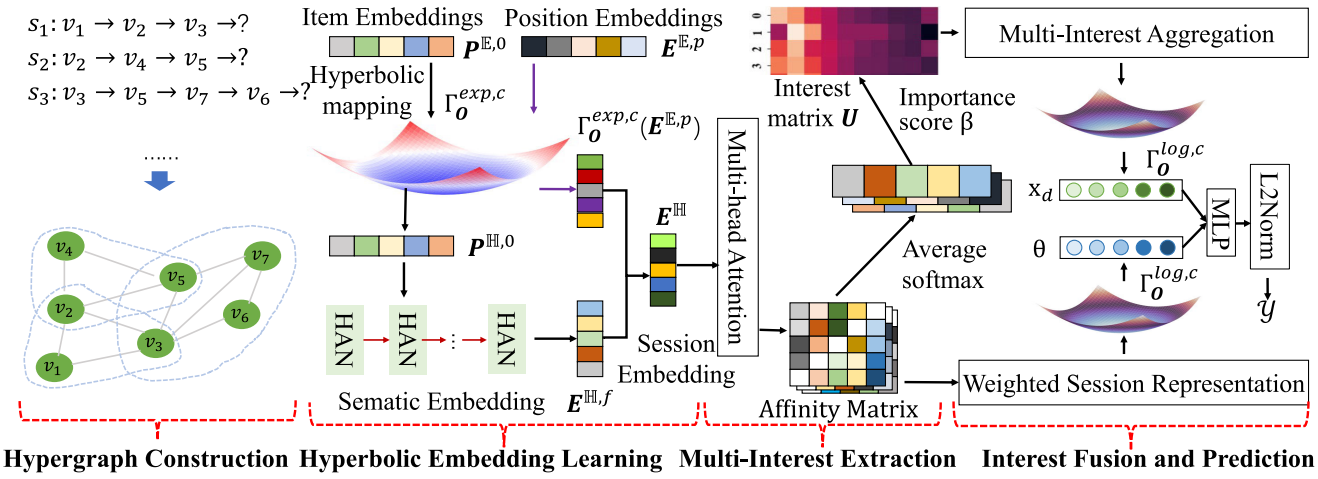


Fig. 2. Workflow of the proposed MIHRN model.

$\Gamma_{\mathbf{o}}^{\log,c}(\Gamma_{\mathbf{o}}^{\exp,c}(\widetilde{\mathbf{P}}_i^{\mathbb{E},0})) = \widetilde{\mathbf{P}}_i^{\mathbb{E},0}$, which is calculated as

$$\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{P}_i^{\mathbb{H},0}\right) = d_{\mathcal{L}}\left(o, \mathbf{P}_i^{\mathbb{H},0}\right) \frac{\mathbf{P}_i^{\mathbb{H},0} + \frac{1}{c}\left\langle \mathbf{o}, \mathbf{P}_i^{\mathbb{H},0}\right\rangle_{\mathcal{L}} \mathbf{o}}{\left\| \mathbf{P}_i^{\mathbb{H},0} + \frac{1}{c}\left\langle \mathbf{o}, \mathbf{P}_i^{\mathbb{H},0}\right\rangle_{\mathcal{L}} \mathbf{o} \right\|_{\mathcal{L}}} \tag{2}$$

where $||\mathbb{H}||_{\mathcal{L}}$ is the Lorentzian norm, and d is the item dimension in both the hyperbolic space and tangent space. The distance $d_{\mathcal{L}}(\mathbf{o}, \mathbf{P}_i^{\mathbb{H},0})$ between the reference point $\mathbf{o}$ and hyperbolic embedding $\mathbf{P}_i^{\mathbb{H},0}$ is calculated as following:

$$d_{\mathcal{L}}\left(o, \mathbf{P}_i^{\mathbb{H},0}\right) = \sqrt{c}\operatorname{arcosh}\left(-\frac{\left\langle \mathbf{o}, \mathbf{P}_i^{\mathbb{H},0}\right\rangle_{\mathcal{L}}}{c}\right). \tag{3}$$

b) Semantic embedding: Using the initial hyperbolic embedding matrix $\mathbf{P}^{\mathbb{H},0}$, a hypergraph attention network (HAN) was further used to extract the item's hyperbolic semantic features by exploiting the high-order relationships and local clustering structure within the hypergraph. A crucial aspect of applying convolutional operations on a hypergraph lies in determining the method for calculating the transition probability between two vertices, which facilitates the propagation of vertex embeddings within the hypergraph. Following the works of [21] and [50], and under the assumption that more propagations should occur between vertices connected by a common hyperedge, and hyperedges with larger weights deserve more confidence in the propagation [54], the hypergraph convolutional operation in hyperbolic space is defined as follows:

$$\mathbf{P}_i^{\mathbb{H},L} = \Gamma_{\mathbf{X}_i^{\mathbb{H},L}}^{\exp,c}\left(\sum_{j\in\mathcal{N}(i)} \mathbf{U}_{ij}\Gamma_{\mathbf{X}_i^{\mathbb{H},L}}^{\log,c}\left(\mathbf{X}_j^{\mathbb{H},L}\right)\right) \tag{4}$$

$$\mathbf{U}_{ij} = \frac{\exp\left(\mathrm{MLP}\left(\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{X}_i^{\mathbb{H},L}\right)||\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{X}_j^{\mathbb{H},L}\right)\right)\right)}{\sum_{k\in\mathcal{N}(i)}\exp\left(\mathrm{MLP}\left(\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{X}_i^{\mathbb{H},L}\right)||\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{X}_k^{\mathbb{H},L}\right)\right)\right)} \tag{5}$$

$$\mathbf{X}_i^{\mathbb{H},L} = \Gamma_{\mathbf{o}}^{\exp,c}\left(\mathbf{W}^L\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{P}_i^{\mathbb{H},L-1}\right)\right) \oplus^c \mathbf{b}^L \tag{6}$$

where $\mathbf{P}_i^{\mathbb{H},L}$ is the aggregated hyperbolic embedding of item v_i after L layer ($L \geq 1$) convolutional operation; $\mathbf{X}_i^{\mathbb{H},L}$ is the hyperbolic hidden embedding of item v_i in the Lth layer via linear transformation, with $\mathbf{W}^L$ and $\mathbf{b}^L$ denoting the weight and bias operation, respectively. In addition, $\oplus^c$ is the hyperbolic bias addition, which is calculated according to the work [55]. According to [47], $\mathbf{X}_i^{\mathbb{H},L}$ was used as the north point for hyperbolic mapping. $\mathcal{N}(i)$ represents the neighboring nodes of item v_i with the same hyperedge as v_i. $\mathbf{U}_{ij}$ is the dynamic transition probability, which reveals the intrinsic relationship between items. To fully exploit higher order relationships, the denseGCN architecture was used due to its superior performance compared with PlainGCN, SkipGCN, and ResGCN, as demonstrated in our experiments. Then, the hyperbolic embedding of all the items is denoted as matrix $\mathbf{P}^{\mathbb{H},L} \in \mathbb{R}^{|\mathcal{V}|\times d}$.

c) Session embedding: For each item v_i in the current session s, a feature vector $\mathbf{X}_i \in \mathbb{R}^{1\times|\mathcal{V}|}$ is created using one-hot encoding technique, and all the items in the session s are transformed into matrices $\mathbf{X} \in \mathbb{R}^{T\times|\mathcal{V}|}$. Subsequently, this is mapped to the hyperbolic embedding matrix $\mathbf{E}^{\mathbb{H},f} \in \mathbb{R}^{T\times d}$, generated by $\mathbf{E}^{\mathbb{H},f} = \mathbf{X}\mathbf{P}^{\mathbb{H},L}$. In addition, considering that items at different positions in the session sequence carry distinct significance and to better use the order information of the session, reverse-position embedding was introduced into the hyperbolic embedding following the work [5]. Based on the length of the session s, the reversed position embedding matrix $\mathbf{E}^{\mathbb{E},p} \in \mathbb{R}^{T\times d}$ in the Euclidean space is obtained. The exponential mapping is used to map the position embedding from its tangent space to the Poincaré ball, resulting in the final hyperbolic embedding matrix $\mathbf{E}^{\mathbb{H}} \in \mathbb{R}^{T\times 2d}$ by concatenation operation

$$\mathbf{E}^{\mathbb{H}} = \left[\mathbf{E}^{\mathbb{H},f} \| \Gamma_{\mathbf{o}}^{\exp,c}\left(\mathbf{E}^{\mathbb{E},p}\right)\right]. \tag{7}$$

3) Multiaspect Interest Extractor: The core component in session recommendation is modeling the user's interests, which takes a session as input and outputs a user representation $\mathbf{u}$ summarizing the user's interest information in the current session. Compressing session information into a single

representation vector through a fixed embedding scheme can be a bottleneck for session representation. We argue that a user's interest in a session is typically diverse, and not all the items in the session are relevant to the user's current preference. Thus, we propose to learn multiple-aspect interest representation to express the diversity of the user's interests.

Inspired by the work of [56], we extract M interest vectors from a given session through an attention mechanism. Specifically, we introduce M learnable affinity matrices, denoted as $\mathbf{C}^{(1)}, \mathbf{C}^{(2)}, \ldots, \mathbf{C}^{(M)}$, where each $\mathbf{C}^{(m)}$ extracts a specific interest vector $\mathbf{u}^{(m)}$ through attention over the session. Analogous to linear transformation in the Euclidean space, the hyperbolic embedding $\mathbf{E}^{\mathbb{H}}$ of each session is transformed into different hyperbolic semantic spaces via hyperbolic linear transformation according [55], denoted as hyperbolic queries $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$. By splitting them into M aspects, the affinity matrix $\mathbf{C}^{(m)}$ of the mth aspect is calculated as

$$\mathbf{C}^{(m)} = \frac{\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{Q}^{(m)}\right)\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{K}^{(m)}\right)^{T}}{\sqrt{2d/M}} \tag{8}$$

where $\mathbf{Q}^{(m)}$ and $\mathbf{K}^{(m)}$ are in $\mathbb{R}^{T\times 2d/M}$. Then, following the approach in [40], the mth interest can be obtained by a weighted average operation

$$\mathbf{u}^{(m)} = \Gamma_{\mathbf{o}}^{\exp,c}\left(\sum_{i=1}^{T}\alpha_i^{(m)}\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{E}_i^{\mathbb{H}}\right)\right) \tag{9}$$

$$\alpha_i^{(m)} = \sigma\left(\frac{1}{M}\sum_{j=1,\ j\neq i}^{T}\mathbf{C}_{ij}^{(m)}\right) \tag{10}$$

where $\alpha_i^{(m)}$ represents the normalized importance weight of each item v_i for the session under the mth affinity matrix. In this manner, we can generate the user's multiple interest matrix $\mathbf{U} = [\mathbf{u}^{(1)}, \mathbf{u}^{(2)}, \ldots, \mathbf{u}^{(M)}]$, with each row representing a certain aspect of the user's interest.

4) Interest Fusion and Prediction:

a) Generating weighted session representation: To capture the high-level interaction patterns among items in the session and enhance the model's nonlinearity ability to obtain reliable session representation, a weighted session representation module was designed. This module takes into account the user's current diverse interest features and distinguishes the importance of each behavior. First, the weighted hidden hyperbolic representation of all the items in session s under M affinity matrices was generated

$$\mathbf{H} = \left[\mathbf{H}^{(1)}||\ \mathbf{H}^{(2)}||\ldots||\mathbf{H}^{(K)}\right] \tag{11}$$

$$\mathbf{H}^{(m)} = \Gamma_{\mathbf{o}}^{\exp,c}\left(\sigma\left(\mathbf{C}^{(m)}\right)\Gamma_{\mathbf{o}}^{\log,c}(\mathbf{V})\right) \tag{12}$$

where $\|$ represents the concatenation operation, $\sigma(\cdot)$ is the softmax function, and $\mathbf{H}\in\mathbb{R}^{L\times 2dM}$. Afterward the high-order feature interaction patterns among different latent dimensions were obtained by a two-layer feed-forward network (FFN) with dropout masking and residual connection techniques

$$\mathbf{X} = \Gamma_{\mathbf{o}}^{\exp,c}\Bigg(\mathrm{LReLU}\Bigg(\mathbf{W}_2\cdot\mathrm{LReLU}\Big(\mathbf{W}_1\Gamma_{\mathbf{o}}^{\log,c} \times\left(\left[\widetilde{\mathbf{H}}\|\mathbf{x}_d\right]\right)+\mathbf{b}_1\Big)\Bigg)+\mathbf{b}_2\Bigg) \tag{13}$$

$$\widetilde{\mathbf{H}} = \Gamma_{\mathbf{o}}^{\exp,c}\left(\mathrm{Dropout}\left(\Gamma_{\mathbf{o}}^{\log,c}(\mathbf{H})\right)\ +\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{E}^{\mathbb{H}}\right)\right) \tag{14}$$

$$\mathbf{x}_d = \Gamma_{\mathbf{o}}^{\exp,c}\Bigg(W_d\Big[\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{u}^{(1)}\right)\|\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{u}^{(2)}\right)\|\ldots \times\|\Gamma_{\mathbf{o}}{}^{\log,c}\left(\mathbf{u}^{(K)}\right)\Big]+\mathbf{b}_d\Bigg) \tag{15}$$

where $\mathbf{x}_d$ is the aggregated user interest; $\mathbf{W}_1$, $\mathbf{W}_2\in\mathbb{R}^{2d\times 2d}$, $\mathbf{W}_d\in\mathbb{R}^{2d\times 2Md}$, and $\mathbf{b}_1$, $\mathbf{b}_2$, $\mathbf{b}_d\in\mathbb{R}^{2d}$ are learnable parameters. Dropout masking and residual connection are used to prevent overfitting and enhance attention representation ability. $\|$ represents the concatenation operation. Dropout and residual connection are further used on the output of the FFN to enhance nonlinearity ability, denoted as $\widetilde{\mathbf{X}}\in\mathbb{R}^{(T+1)\times 2d}$. Then, we consider $\widetilde{\mathbf{X}}_{T+1}$ as the query, $\widetilde{\mathbf{X}}_{1:T}$ as the key, and the nonlinear transformation of the initial item embedding $\mathbf{E}^{\mathbb{H}}$ as a value. Based on previous studies [12], [27], the weighted session representation was generated

$$\theta = \Gamma_{\mathbf{o}}^{\exp,c}\left(\sum_{i=1}^{T}\gamma\left(\mathbf{W}_{\widetilde{v}}\Gamma_{\mathbf{o}}^{\log,c}\left(\mathbf{E}^{\mathbb{H}}\right)+\mathbf{b}_{\widetilde{v}}\right)\right) \tag{16}$$

$$\gamma = \mathrm{Softmax}\Big(\mathbf{f}^{T}\mathrm{LReLU}\Big(\mathbf{W}_3\Gamma_{\mathbf{o}}^{\log,c}\left(\widetilde{\mathbf{X}}_{1:T}^{T}\right) +\mathbf{W}_4\Gamma_{\mathbf{o}}^{\log,c}\left(\widetilde{\mathbf{X}}_{T+1}\right)+\mathbf{c}\Big)\Big) \tag{17}$$

where $\mathbf{W}*\in\mathbb{R}^{2d\times 2d}$ and $\mathbf{W}\widetilde{v}\in\mathbb{R}^{2d\times 2d}$ are the weighting matrices; $\mathbf{c}\in\mathbb{R}^{2d}$ and $\mathbf{b}_{\widetilde{v}}\in\mathbb{R}^{2d}$ are the bias vectors.

b) Next-item recommendation: By aggregating the user's multiaspect interest and session representation using FFN, we can acquire the hybrid session representation $\mathbf{z}$

$$\mathbf{z} = \mathrm{Dropout}\Big(\mathrm{SeLU}\Big(\mathbf{W}_z\ \Big[\Gamma_{\mathbf{o}}^{\log,c}(\mathbf{x}_d)||\Gamma_{\mathbf{o}}^{\log,c}(\theta)\Big]+\mathbf{b}_z\Big)\Big) \tag{18}$$

where $\mathbf{W}_z\in\mathbb{R}^{4d\times d}$ and $\mathbf{b}_3\in\mathbb{R}^{d}$ are trainable parameters for generating the hybrid session representation, and SeLU [57] is the activation function that introduces nonlinearity to the hybrid session representation. According to [58], L2 normalization can stabilize the training process and accelerate the model convergence. Therefore, we normalize the hybrid session representation and the embedding of all the items and use the inner product with softmax to calculate the similarity scores for predicting the click probability

$$\hat{\mathbf{z}} = \mathbf{W}_p\mathrm{L2Norm}(\mathbf{z})\ \ \hat{\mathbf{e}}_i = \mathrm{L2Norm}\Big(\ \Gamma_{\mathbf{o}}^{\log,c}\Big(\ \mathbf{E}_i^{\mathbb{H}}\Big)\Big) \tag{19}$$

$$y_i = \mathrm{Softmax}\left(\hat{\mathbf{z}}^{T}\hat{\mathbf{e}}_i\right) \tag{20}$$

where $\mathbf{E}_i^{\mathbb{H}}$ denotes the embedding of the ith candidate item in $\mathcal{V}$, L2Norm$(\cdot)$ represents the L2 normalization function, $\mathbf{W}_p$ is the normalized weight [27], and y_i is the final prediction score of item v_i. Finally, the cross-entropy loss was used as the loss function

$$\mathcal{L}_{\mathrm{main}} = -\sum_{i=1}^{|\mathcal{V}|}\widehat{y_i}\log(y_i)+(1-\widehat{y_i})\log(1-y_i) \tag{21}$$

where $\hat{y}$ denotes the one-hot encoding vector of the ground-truth item.

C. Enhanced Learning and Optimization

1) Enhanced Learning: Considering that the cross-entropy loss function may overemphasize the model's performance and lead to overfitting due to the sparsity of the session data, we design a session enhancement learning scheme using dropout masking operations. This scheme constructs different views of the same session while still maintaining the main preferences hidden in historical behaviors. Specifically, for each session s, we generate two different session representations through (8)–(16) twice by adopting different dropout masks independently, denoted as $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$. Thereafter, the remaining session representations are used in the mini-batch for negative pairs, and the normalized temperature-scaled cross-entropy loss (NT-Xent) between the positive and negative pairs is adopted as the contrastive learning objective [59]

$$\mathcal{L}_{\text{el}} = \frac{1}{N}\sum_{i}^{N} \log \frac{e^{\text{sim}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_i^*)/\tau}}{\sum_{j=1}^{N} e^{\text{sim}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j^*)/\tau} + \sum_{j=1, j\neq i}^{N} e^{\text{sim}(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j^*)/\tau}} \tag{22}$$

where $\boldsymbol{\theta}_i$ is the standard session representation for session i in our model, and $\boldsymbol{\theta}_i^*$ is the mirror session embedding for session i in positive pairs. N denotes the batch size of each mini-batch, and τ controls the temperature. $\text{sim}(\cdot) : \mathbb{R}^{2d} \times \mathbb{R}^{2d} \rightarrow \mathbb{R}$ is the discriminator function that takes two vectors as its input and outputs their similarity scores. For simplicity, cosine similarity was used in our model. During each training step, the discriminator $\text{sim}(\cdot)$ was used to distinguish the mirror session embedding and other session embeddings (including original and mirror embeddings) in that batch.

2) Joint Optimization: The joint learning framework with AdamW [60] algorithm was used for model training

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \beta \mathcal{L}_{\text{el}} \tag{23}$$

where $\mathcal{L}_{\text{main}}$ and $\mathcal{L}_{\text{el}}$ denote the loss of the main task and enhanced learning, respectively, and β denotes the hyperparameter used to control the magnitude of the enhanced learning task. In this article, β was set to 0.4 based on the highest performance observed in our experiments.

IV. Experiments

We conduct extensive experiments to prove the efficacy of our MIHRN model. The experimental settings, including datasets, baseline approaches, evaluation metrics, and hyperparameters setting are first introduced, and then the experimental results and discussion are presented in detail.

A. Experimental Settings

1) Datasets: Three publicly available datasets, Diginetica,[1] Tmall,[2] and Nowplaying,[3] were used to validate the effectiveness of the proposed model. The Diginetica dataset was obtained from the CIKM Cup 2016, containing only transitional data. The Tmall dataset was obtained from the IJCAI-15 competition and consists of anonymized user shopping logs on the Tmall online shopping platform. The Nowplaying dataset describes the music listening behavior of users. Following the works of SrGNN [3] and DHCN [17], sessions containing only one item and items appearing fewer than five times were removed from the datasets. Subsequently, to evaluate the proposed model, following the settings in the literature [3], [17], the datasets were split into training and testing sets. In addition, similar to the approaches in the literature [3], [14], we augmented and labeled the dataset using a sequence splitting method. For example, for an input session $s = [v_{s,1}, v_{s,2}, v_{s,3}, \ldots, v_{s,t}]$, we generated a series of sessions and labels: $([v_{s,1}], v_{s,2})$, $([v_{s,1}, v_{s,2}], v_{s,3})$, $([v_{s,1}, v_{s,2}, v_{s,3}], v_{s,4})$, and so on, until $([v_{s,1}, v_{s,2}, v_{s,3}, \ldots, v_{s,t-1}], v_{s,t})$, where $[v_{s,1}, v_{s,2}, v_{s,3}, \ldots, v_{s,i-1}]$ represents the generated session and $v_{s,i}$ denotes the item that the user may click next in the current session. Table I presents the statistics of the datasets.

TABLE I
Dataset Statistics

Dataset	#training	#testing	#items	Avg.len
Diginetica	719,470	60,858	43,097	5.12
Tmall	351,268	25,898	40,728	6.69
Nowplaying	825,304	89,824	60,417	7.42

2) Comparative Approaches: The MIHRN model was compared with the following state-of-the-art approaches that have widely been used as points of comparison in previous studies.

1) *Item-KNN [37]:* Uses cosine similarity to calculate the similarity between candidate items and recommend items that are similar to the previous items in the session.
2) *FPMC [6]:* A hybrid sequential model combining matrix factorization and first-order Markov chains.
3) *GRU4REC [3]:* Uses a session-parallel mini-batch training process and adopts a ranking-based loss function to model the user sequence.[4]
4) *NARM [14]:* Combines an RNN with an attention mechanism to model users' sequential behaviors and capture their main purpose.[5]
5) *STAMP [15]:* Uses an attention network to capture a user's general interest and models the current interest based on the last clicked item.[6]
6) *SRGNN [3]:* Applies a gated GNN to learn the item representation and uses the last clicked item for the final recommendation.[7]
7) *S^2-DHCN [17]:* Uses hypergraph convolutional networks to capture high-order relationships and enhances the model with self-supervised learning and line graphs.[8]
8) *GCE-GNN [12]:* Encodes item and session embedding from both global level and session level to improve the recommendation.[9]

[1] http://cikm2016.cs.iupui.edu/cikm-cup
[2] https://tianchi.aliyun.com/dataset/dataDetail?dataId=42
[3] http://dbis-nowplaying.uibk.ac.at/#nowplaying
[4] https://github.com/hidasib/GRU4Rec
[5] https://github.com/lijingsdu/sessionRec_NARM
[6] https://github.com/uestcnlp/STAMP
[7] https://github.com/CRIPAC-DIG/SR-GNN
[8] https://github.com/xiaxin1998/DHCN
[9] https://github.com/CCIIPLab/GCE-GNN

9) *COTREC [61]:* Combines self-supervised learning with co-training to select evolving pseudolabels as informative self-supervision examples for each view iteratively, thereby improving the recommendation.[10]
10) *Disen-GNN [62]:* Uses disentangled learning to decompose item embeddings into multiple factors and then apply a gated GNN (GGNN) to learn these factor embeddings using the item adjacency similarity matrix for each factor.[11]
11) *Attent-Mixer [32]:* Leverages both concept-view and instance-view readouts to enable multilevel reasoning over item transitions.[12]
12) *TASI-GNN [42]:* Integrates GNNs with sparse graph attention mechanisms and a similar-intent collaboration module to learn enhanced, denoised target information and long-term preferences, ensuring a reliable session representation.[13]
13) *HADCG [22]:* Converts sessions into hyperbolic session graphs to integrate chronological and hierarchical information.

These comparative approaches can be divided into three paradigms: traditional approaches, such as Item-KNN and FPMC; RNN-based approaches, including pure RNN-based approaches (i.e., GRU4REC) and hybrid approaches (i.e., NARM and STAMP); and GNN-based approaches, including SRGNN, S2-DHCN, GCE-GNN, COTREC, Disen-GNN, Attent-Mixer, TASI-GNN, and HADCG. Note that the Attent-Mixer and TASI-GNN are newly published multi-interest session representation models.

3) Evaluation Metrics: Following the methodologies outlined in [17] and [12], we used two commonly used evaluation metrics: precision at N (P@N) and mean reciprocal rank at N (MRR@N), where N was set to 10 and 20. P@N is a widely used metric for assessing predictive accuracy, representing the proportion of correctly recommended items within the top-N recommendations, without considering the ranking order of the recommended items. The P@N score was computed as

$$\text{P@N} = \frac{\#\text{Hit}}{\#\text{Test}} \tag{24}$$

where #Test denotes the number of sessions in the test set, and #Hit denotes the number of cases in which the desired items are in the top-N ranking lists. MRR@N is the average of the reciprocal ranks of the desired item, which evaluates the ranked recommended items. MRR@N was calculated as

$$\text{MRR@N} = \frac{1}{\#\text{Test}} \sum_{v_{\text{label}} \in S_{\text{test}}} \frac{1}{\text{Rank}(v_{\text{label}})} \tag{25}$$

where v_{label} denotes the desired item. The reciprocal rank is set to zero when the rank exceeds N. Because MRR@N considers the order of recommendation ranking, a large value indicates that the recommend items are at the top of the ranking list.

[10]https://github.com/xiaxin1998/COTREC
[11]https://github.com/AnsongLi/Disen-GNN
[12]https://github.com/Peiyance/Atten-Mixer-torch
[13]https://github.com/kkkk889966332211/TASI-GNN

TABLE II
OPTIMAL HYPERPARAMETERS ON EACH DATASET

Hyper-parameters	Datasets		
	Tmall	Diginetica	Nowplaying
Batch Size	512	512	512
Droupout Rate p	0.3	0.2	0.2
Number of Layers L	5	4	5
Number of Headers K	8	5	8
τ	0.1	0.1	0.7

4) Hyperparameters Settings: Following previous studies [3], [14], and the embedding dimension d is set to 100 for all the comparative approaches. For our MIHRN model, the batch size was set to 512, normalized weight w_k was set to 20, and the AdamW [60] optimizer with a learning rate of 0.0001 was used. The optimal hyperparameters for MIHRN on the Tmall, Diginetica and Nowplaying datasets are listed in Table II. In addition, for those comparative approaches, we referred to the best hyperparameter setups reported in the original papers. All the experiments were conducted on a server machine equipped with an NVIDIA RTX 6000 24-GB GPU.

B. Results and Discussion

1) Results With Comparative Approaches: The experimental results of the MIHRN model and the comparative approaches on the Tmall, Diginetica, and Nowplaying datasets are shown in Table III, with the best results for each column highlighted in boldface and the second-best results underlined. The following conclusions can be drawn from the results.

1) RNN-based approaches, such as GRU4REC, demonstrate higher performance compared with traditional approaches such as Item-KNN and FPMC. This highlights the importance of leveraging deep learning techniques to explore sequential dependencies within sessions. NARM and STAMP exhibit substantial performance improvements over GRU4REC, indicating that user behavior does not strictly adhere to sequential dependencies as seen in natural language. The incorporation of attention mechanisms in NARM and STAMP enables capturing the main intention by discerning irrelevant interactions within sessions.
2) GNN-based methods such as SR-GNN, S^2-DHCN, GCE-GNN, DSAN, COTREC, Disen-GNN, Attent-MiXer, TASI-GNN, and HADCG significantly improve recommendation accuracy by effectively modeling high-order item transitions. SR-GNN excels over earlier models, showcasing the advanced ability of GNNs to manage complex interactions. S^2-DHCN achieves strong results using a hypergraph neural network to capture high-order relationships among items, leveraging inter-session information, and incorporating self-supervised learning. However, the use of hyperedges, which connect multiple nodes, results in a sparser graph structure and introduces additional noise. GCE-GNN enhances performance by learning item embeddings from both session and global graphs, providing better inference of

TABLE III
RESULTS OF MIHRN WITH COMPARATIVE APPROACHES ON THREE DATASETS

Method	Tmall				Diginetica				Nowplaying			
	P@10	M@10	P@20	M@20	P@10	M@10	P@20	M@20	P@10	M@10	P@20	M@20
Item-KNN	6.65	3.11	9.15	3.31	25.07	10.77	35.75	11.57	10.96	4.55	15.94	4.91
FPMC	13.10	7.12	16.06	7.32	15.43	6.20	26.53	6.95	5.28	2.68	7.36	2.82
GRU4REC	9.47	5.78	10.93	5.89	17.93	7.33	29.45	8.33	6.74	4.40	7.92	4.48
NARM	19.17	10.42	23.30	10.70	35.44	15.13	49.70	16.17	13.60	6.62	18.59	6.93
STAMP	22.63	13.12	26.47	13.36	33.98	14.26	45.64	14.32	13.22	6.57	17.66	6.88
SR-GNN	23.41	13.45	27.57	13.72	36.86	15.52	50.73	17.59	14.17	7.15	18.87	7.47
S^2-DHCN	26.22	14.60	31.42	15.05	40.21	17.59	53.66	18.51	17.35	7.87	23.04	8.18
GCE-GNN	28.01	15.08	33.42	15.42	41.16	18.15	54.22	19.04	16.94	8.03	22.37	8.40
COTREC	30.62	17.65	36.35	18.04	41.88	18.16	54.18	19.07	17.32	7.53	23.51	7.96
Disen-GNN	24.21	14.63	28.97	14.93	40.61	18.08	53.79	18.99	15.93	8.04	22.22	8.22
Attent-MiXer	29.27	16.55	34.98	16.72	41.47	17.94	55.12	18.90	16.57	8.12	22.00	8.49
TASI-GNN	30.67	18.81	34.71	19.08	46.89	21.72	59.08	22.56	14.10	6.98	19.05	7.53
HADCG	32.38	18.04	37.29	18.95	43.35	19.14	56.21	20.36	21.17	10.63	26.72	12.45
MIHRN	**40.09**	**19.76**	**47.11**	**20.26**	**49.54**	**21.98**	**61.36**	**22.79**	**28.97**	**11.91**	**36.99**	**12.47**
Improv.	**23.81%**	**9.53%**	**26.33%**	**6.91%**	**14.28%**	**14.84%**	**9.16%**	**11.94%**	**36.84%**	**12.04%**	**38.44%**	**0.16%**

user preferences. Disen-GNN uses disentangled GNNs to create embeddings based on item adjacency similarity matrices for each factor, performing similar to S^2-DHCN. Yet, its focus on the last item may lead to noisy and inaccurate session embeddings. AttenMixer suggests that complex GNN propagation might be redundant in some contexts, boosting performance by applying multilevel user intentions for item transition reasoning but neglecting the rich collaborative information between sessions. TASI-GNN, using an adaptive denoising objective, effectively learns session representations from items, surpassing previous models but lacking in capturing high-order feature representations in intricate graph data. In addition, by modeling temporal and spatial feature structures in hyperbolic space, HADCG achieves significant performance improvements on the Tmall and Nowplaying datasets except for the M@10 and M@20 metric on the Tmall dataset.

3) Our proposed approach, MIHRN, achieves substantial improvements and consistently outperforms comparative approaches across all the datasets, demonstrating the effectiveness of the MIHRN model. Several factors contribute to this performance improvement. First, modeling high-order hierarchical spatial structures of item popularity in session data using hyperbolic geometry enables depicting more discriminative distribution of session representation. Second, capturing multiaspect interest using affinity matrices and generating weighted session representation through higher order feature interactions enhances recommendation accuracy. By skillfully modeling complex higher order hierarchical spatial structures among items in hyperbolic space and capturing user preferences from multiple perspectives, our model achieves superior results. Moreover, incorporating enhance learning further improves session representation learning. Furthermore, compared with the Tmall and Diginetica datasets, where users generally have specific targets for each behavior, users' music listening behaviors are typically random. While graph-based models may recall more items, the ranking of recalled items is not accurate. In contrast, the MIHRN model achieves considerably more precise rankings due to the adoption of multilayer attention to calculate the correlation between each item for recommendation.

2) Ablation Study: Our MIHRN model comprises several innovative components, and to further verify the contribution of each component to the final recommendation performance, we designed several variants of MIHRN and evaluated them on the three datasets using the P@10 and M@10 metrics: w/o HE: This variant removes the hyperbolic embedding module and instead uses initialized Euclidean embeddings. w/o MI: This variant removes the multiaspect interest learning module and uses the last clicked item in the current session as the user's interest representation. w/o EL: This variant removes the enhance learning module, and only the cross-entropy loss is used for model optimization. The comparison results are reported in Table IV. It is observed that the performance of the w/o HE variant is the lowest, indicating that learning the complex hierarchical structures inherent in session data is crucial, and this can be effectively achieved in hyperbolic geometry space. This is especially important for SBR tasks, which are characterized by sparse, power-law distributed data. Comparing with the MIHRN model, the performance of the w/o MI variant decreases to a certain extent, validating that the extracted multiaspect interest can reveal the user's overall preference. Furthermore, the w/o EL variant performs better than the w/o MI variant, demonstrating the significance of the enhance learning scheme in enhancing the recommendation performance. This emphasizes the importance of leveraging enhance learning techniques to construct different views of the same session, thereby maintaining the main preferences hidden in historical behaviors.

3) Training Time Comparison: We also conducted experiments to compare the training times of the MIHRN model with the existing self-supervised learning-based models, namely, the S^2-DHCN and COTREC models, to validate the computational efficiency of our model. Fig. 3(a) depicts the time required for each model to train one epoch, while Fig. 3(b)

TABLE IV
RESULTS OF ABLATION STUDY

Datasets	Tmall		Diginetica		Nowplaying	
	P@10	M@10	P@10	M@10	P@10	M@10
MIHRN	**40.09**	**19.76**	**49.54**	**21.98**	**28.97**	**11.91**
w/o HE	32.41	17.57	41.08	18.32	17.12	8.65
w/o MI	37.56	17.38	46.74	19.95	24.37	10.12
w/o EL	38.81	19.26	46.58	20.76	25.80	11.01

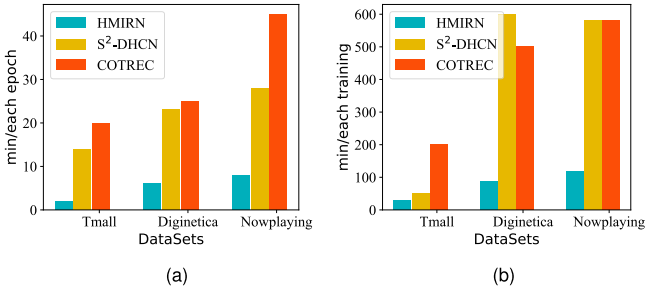


Fig. 3. Comparison of training times. (a) Time consumption for each epoch. (b) Time consumption for each training.

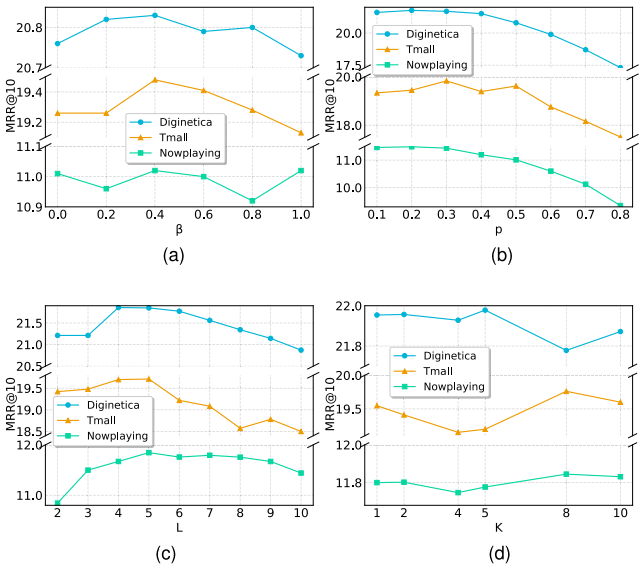


Fig. 4. Results of hyperparameters' analysis. (a) Balancing factor β. (b) Dropout rate p. (c) Number of layers L. (d) Numbers of heads K.

illustrates the overall training time for each model. It is evident from the figures that the training time of our model is significantly less than that of the S^2-DHCN and COTREC models, while achieving better performance than both. This is primarily attributed to the fact that the latter two models use complex encoders to capture different views of the session, whereas the MIHRN model learns hierarchy structures and multiaspect interest in hyperbolic space using two simple modules. By leveraging these modules, the MIHRN model captures intrinsic factors without increasing model complexity, thereby improving model effectiveness while reducing training time.

4) Hyperparameter Analysis: We conducted detailed hyperparameter experiments on three datasets to explore the impact of various hyperparameters in the MIHRN model. In the following experiments, we changed only one parameter at a time, while keeping the optimal settings for each dataset unchanged.

a) Impact of beta β: In the MIHRN model, β controls the effect of divergence loss. To explore the overall impact of the similarity constraint task, we set the number of heads $K = 8$, the number of layers $L = 3$, and the dropout rate $p = 0.5$. We trained the model with different values of β in the set $\{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$. The M@10 results obtained are shown in Fig. 4(a). It can be observed that the M@10 results for all the three datasets reach their optimal values when $\beta = 0.4$. Subsequently, as the β value increases, M@10 for the three datasets consistently decreases. We believe this is because excessively large dissimilarity between two modules makes it challenging for them to supervise each other effectively.

b) Impact of dropout rate p : Dropout regularization techniques are used to create positive pairs for contrastive learning and prevent the proposed model from overfitting. The key idea for dropout is to randomly mask neurons with probability p during training, while setting $p = 0$ during testing. Thus, dropout layers can force the model to learn a more robust representation. By setting the number of heads $K = 8$, the number of layers $L = 3$, and varying the dropout rate p in $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$, Fig. 4(b) presents the results of using the M@10 metrics on the Tmall, Diginetica, and Nowplaying datasets. The results show that the model achieves the best performance when the dropout ratio is set to 0.3 on the Tmall dataset, and 0.2 on the Diginetica and Nowplaying datasets. However, the performance starts to deteriorate when the dropout rate is increased because the model becomes difficult to train with limited available neurons, and the contrastive learning discriminator struggles to distinguish between positive and negative pairs.

c) Impact of layer numbers L: Intuitively, increasing the number of layers allows the GNN to better capture the complex relationships between nodes, thereby improving the model's expressive power. However, this does not imply that the model's performance will linearly increase with the number of layers. Due to the oversmoothing issue and the computational cost associated with exponential growth, excessively deep GNNs may actually degrade model performance. By fixing the optimized dropout rate p and setting the number of heads $K = 8$, we varied the number of layers L in $\{2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Fig. 4(c) presents the performance of the M@10 results under different layer numbers L on the three datasets. The results show that the model achieves the highest M@10 values when L is set to 5 for the Tmall and Nowplaying datasets and set to 4 for the Diginetica dataset. With further increases in L, the M@10 values for the three datasets consistently decline.

d) Impact of attention heads K: The attention mechanism plays a critical role in SBR tasks as it focuses on important items while ignoring less important ones by applying different attention weights. Moreover, multihead attention allows the model to attend to information from different representation subspaces with different attention heads in

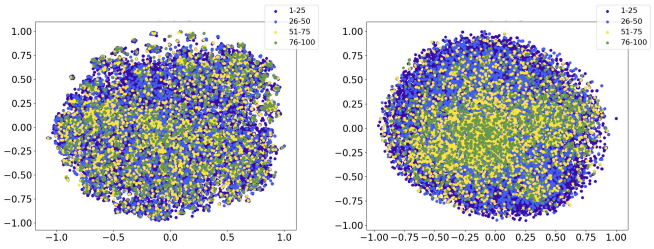


Fig. 5. t-SNE visualizations of item embeddings from the Nowplaying dataset. (a) Euclidean embedding. (b) Hyperbolic embedding.

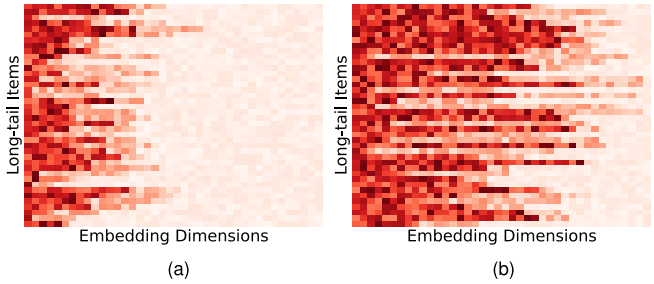


Fig. 6. Visualization of item embeddings with fewer than five clicks from the Nowplaying dataset. (a) Euclidean space. (b) Hyperbolic space.

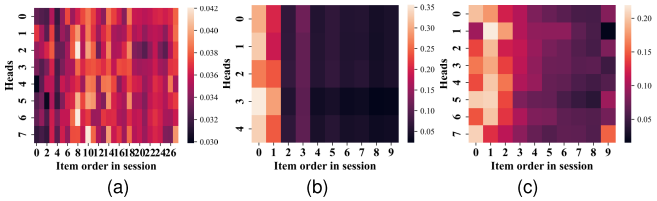


Fig. 7. Visualization of normalized item weight in a session. (a) Tmall. (b) Diginetica. (c) Nowplaying.

parallel. By fixing the optimized dropout rate p and setting the number of layers $L = 3$, we varied the number of heads K in $\{1, 2, 4, 5, 8, 10\}$. Fig. 4(d) presents the performance of the M@10 results under different head numbers K on the three datasets. The results show that MIHRN achieves the best performance when the number of attention heads is set to eight on Tmall and Nowplaying and five on Diginetica. The performance starts to deteriorate when the head number increases beyond the optimal values. This is because more heads produce a smaller subspace, making it difficult to capture all the information, which leads to model overfitting.

5) Case Studies: To further validate that embedding learning in hyperbolic space can capture hierarchical structures in session data with a power-law distribution, we used t-SNE [63] to visualize item embeddings in both hyperbolic and Euclidean spaces. Using the Nowplaying dataset as an example, the visualization results are shown in Fig. 5. In Euclidean space, items of different popularity are randomly distributed without a clear structure. However, in hyperbolic space, hierarchical representation is evident: the most popular items are almost all clustered near the center of the projected latent space, while less popular items are further from the center. This indicates that hyperbolic hypergraphs can learn item embeddings based on the distribution of items in user–item interactions, naturally integrating hierarchical patterns to create more informative item representations. To further validate our model's ability to learn rich representations for items with fewer clicks (fewer than 5), we randomly selected 40 such items from the Nowplaying dataset and visualized their embeddings in both Euclidean and hyperbolic spaces, as shown in Fig. 6. In Euclidean space, only a few dimensions of the embeddings exhibit significant values, suggesting that its limited expressive capacity impedes effective learning and representation of items with few interactions. In contrast, the hyperbolic space, with its superior expressive capability, enables the model to learn more comprehensive representations of these items, demonstrating its effectiveness for this task.

In addition, to intuitively examine whether our model can capture multiple interests, we randomly selected one session from each dataset and visualized the normalized item weights within the session under different attention heads across three datasets, as shown in Fig. 7. The visualization clearly demonstrates that the user's diverse interests are effectively captured by our MIHRN model through the use of different attention heads. Using multiple attention heads, our model can attend to various aspects of the user's preferences simultaneously, highlighting different items in the session that correspond to distinct interests. This multi-interest representation enables the model to provide a more comprehensive understanding of user behavior, leading to better recommendations that align with the user's varied tastes and preferences.

V. Conclusion

With increasing privacy concerns, obtaining comprehensive user interaction data, especially for anonymous users, has become challenging. SBR has emerged as a solution, aiming to predict the next item within an ongoing session without relying on user profiles or historical data. While earlier SBR approaches focused on sequential transfer relationships between adjacent items, modern methodologies incorporate GNNs to capture both local and global item transition patterns, improving SBR performance. However, the existing graph-based methods often overlook the hierarchical, diverse, and variable nature of user interactions, posing challenges in addressing data sparsity and increasing the risk of overfitting. To overcome these limitations, we proposed the MIHRNs, adept at modeling intricate spatial structures and sequence relationships among items in hyperbolic geometry space. By leveraging a hyperbolic hypergraph neural network and a multiaspect interest representation module, MIHRN effectively captures the diverse and variable nature of user interests within sessions. In addition, we introduced a minimal data augmentation strategy to enhance session representation and learning efficiency. Extensive experimental analysis validated MIHRN's superiority in both performance and efficiency. In the future, we aim to design new sequence augmentation methods and encoders better suited to contrastive learning in SBR, further advancing recommendation systems in practical scenarios.

REFERENCES

[1] T. Liu, S. Lou, J. Liao, and H. Feng, "Dynamic and static representation learning network for recommendation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 1, pp. 831–841, Jan. 2024.

[2] D. Jannach and M. Ludewig, "When recurrent neural networks meet the neighborhood for session-based recommendation," in *Proc. 11th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 306–310.

[3] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 346–353.

[4] Z. Liu et al., "Semantic-enhanced contrastive learning for session-based recommendation," *Knowl.-Based Syst.*, vol. 280, Nov. 2023, Art. no. 111001.

[5] G. Shani, D. Heckerman, R. I. Brafman, and C. Boutilier, "An mdp-based recommender system," *J. Mach. Learn. Res.*, vol. 6, no. 9, pp. 1265–1295, 2005.

[6] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World Wide web*, Apr. 2010, pp. 811–820.

[7] Y. K. Tan, X. Xu, and Y. Liu, "Improved recurrent neural networks for session-based recommendations," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, Sep. 2016, pp. 17–22.

[8] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, May 2016, pp. 1–10.

[9] B. Hidasi and A. Karatzoglou, "Recurrent neural networks with top-k gains for session-based recommendations," in *Proc. 27th ACM Int. Conf. Inf. Knowl. Manag.*, Oct. 2018, pp. 843–852.

[10] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

[11] C. Xu et al., "Graph contextualized self-attention network for session-based recommendation," in *Proc. IJCAI*, vol. 19, 2019, pp. 3940–3946.

[12] Z. Wang, W. Wei, G. Cong, X.-L. Li, X.-L. Mao, and M. Qiu, "Global context enhanced graph neural networks for session-based recommendation," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 169–178.

[13] C. Huang et al., "Graph-enhanced multi-task learning of multi-level transition dynamics for session-based recommendation," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 5, pp. 4123–4130.

[14] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, "Neural attentive session-based recommendation," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 1419–1428.

[15] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, "STAMP: Short-term attention/memory priority model for session-based recommendation," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 1831–1839.

[16] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *Proc. IEEE Int. Conf. Data Min. (ICDM)*, Nov. 2018, pp. 197–206.

[17] X. Xia, H. Yin, J. Yu, Q. Wang, L. Cui, and X. Zhang, "Self-supervised hypergraph convolutional networks for session-based recommendation," 2020, *arXiv:2012.06852*.

[18] C. Ding, Z. Zhao, C. Li, Y. Yu, and Q. Zeng, "Session-based recommendation with hypergraph convolutional networks and sequential information embeddings," *Expert Syst. Appl.*, vol. 223, Aug. 2023, Art. no. 119875.

[19] X. Zhang et al., "Price DOES matter! Modeling price and interest preferences in session-based recommendation," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 1684–1693.

[20] X. Zhang, B. Xu, F. Ma, C. Li, Y. Lin, and H. Lin, "Bi-preference learning heterogeneous hypergraph networks for session-based recommendation," *ACM Trans. Inf. Syst.*, vol. 42, no. 3, pp. 1–28, May 2024.

[21] Y. Li et al., "Hyperbolic hypergraphs for sequential recommendation," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 988–997.

[22] J. Su, C. Chen, W. Liu, F. Wu, X. Zheng, and H. Lyu, "Enhancing hierarchy-aware graph networks with deep dual clustering for session-based recommendation," in *Proc. ACM Web Conf.*, Apr. 2023, pp. 165–176.

[23] E. Ravasz and A.-L. Barabási, "Hierarchical organization in complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 67, no. 2, Feb. 2003, Art. no. 026112.

[24] L. Vinh Tran, Y. Tay, S. Zhang, G. Cong, and X. Li, "HyperML: A boosting metric learning approach in hyperbolic space for recommender systems," in *Proc. 13th Int. Conf. Web Search Data Mining*, Jan. 2020, pp. 609–617.

[25] N. Guo et al., "Hyperbolic contrastive graph representation learning for session-based recommendation," *IEEE Trans. Knowl. Data Eng.*, early access, Aug. 29, 2024, doi: 10.1109/TKDE.2023.3295063.

[26] Z. Pan, F. Cai, Y. Ling, and M. de Rijke, "Rethinking item importance in session-based recommendation," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1837–1840.

[27] J. Yuan, Z. Song, M. Sun, X. Wang, and W. X. Zhao, "Dual sparse attention network for session-based recommendation," in *Proc. 35th AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 5, pp. 4635–4643.

[28] Q. Liu, M. Nickel, and D. Kiela, "Hyperbolic graph neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 8230–8241.

[29] Y. Yang et al., "Hyperbolic graph learning for social recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 12, pp. 8488–8501, Dec. 2024.

[30] X. Li, Y. Liu, Z. Liu, and P. S. Yu, "Time-aware hyperbolic graph attention network for session-based recommendation," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2022, pp. 626–635.

[31] J. Guo et al., "Learning multi-granularity consecutive user intent unit for session-based recommendation," in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 343–352.

[32] P. Zhang et al., "Efficiently leveraging multi-level user intent for session-based recommendation via atten-mixer network," in *Proc. 16th ACM Int. Conf. Web Search Data Mining*, Feb. 2023, pp. 168–176.

[33] Q. Tan et al., "Sparse-interest network for sequential recommendation," in *Proc. 14th ACM Int. Conf. Web Search Data Mining*, 2021, pp. 598–606.

[34] S. Zhang et al., "Re4: Learning to re-contrast, re-attend, re-construct for multi-interest recommendation," in *Proc. ACM Web Conf.*, Apr. 2022, pp. 2216–2226.

[35] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 20, 2007, pp. 1257–1264.

[36] Y. Koren, S. Rendle, and R. Bell, "Advances in collaborative filtering," in *Recommender Systems Handbook*. New York, NY, USA: Springer, 2021, pp. 91–142.

[37] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, Apr. 2001, pp. 285–295.

[38] M. Wang, P. Ren, L. Mei, Z. Chen, J. Ma, and M. de Rijke, "A collaborative session-based recommendation approach with parallel memory modules," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2019, pp. 345–354.

[39] A. Luo et al., "Collaborative self-attention network for session-based recommendation," in *Proc. IJCAI*, Jul. 2020, pp. 2591–2597.

[40] R. Qiu, J. Li, Z. Huang, and H. Yin, "Rethinking the item order in session-based recommendation with graph neural networks," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 579–588.

[41] J. Wang, Q. Xu, J. Lei, C. Lin, and B. Xiao, "PA-GGAN: Session-based recommendation with position-aware gated graph attention network," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2020, pp. 1–6.

[42] G. An, J. Sun, Y. Yang, and F. Sun, "Enhancing collaborative information with contrastive learning for session-based recommendation," *Inf. Process. Manage.*, vol. 61, no. 4, Jul. 2024, Art. no. 103738.

[43] S. Wang, L. Hu, Y. Wang, Q. Z. Sheng, M. Orgun, and L. Cao, "Modeling multi-purpose sessions for next-item recommendations via mixture-channel purpose routing networks," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 3771–3777.

[44] M. Choi, H.-Y. Kim, H. Cho, and J. Lee, "Multi-intent-aware session-based recommendation," 2024, *arXiv:2405.00986*.

[45] M. Nickel and D. Kiela, "Poincaré embeddings for learning hierarchical representations," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 6341–6350.

[46] Y. Tan, C. Yang, X. Wei, C. Chen, L. Li, and X. Zheng, "Enhancing recommendation with automated tag taxonomy construction in hyperbolic space," in *Proc. IEEE 38th Int. Conf. Data Eng. (ICDE)*, May 2022, pp. 1180–1192.

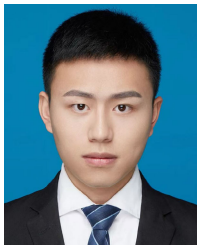
[47] I. Chami, Z. Ying, C. Ré, and J. Leskovec, "Hyperbolic graph convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 4869-4880.

[48] X. Liu, Y. Zhu, and X. Wu, "Joint user profiling with hierarchical attention networks," *Frontiers Comput. Sci.*, vol. 17, no. 3, Jun. 2023, Art. no. 173608.

[49] Y. Zhang, X. Wang, C. Shi, X. Jiang, and Y. Ye, "Hyperbolic graph attention network," *IEEE Trans. Big Data*, vol. 8, no. 6, pp. 1690–1701, Dec. 2022.
[50] J. Sun, Z. Cheng, S. Zuberi, F. Pérez, and M. Volkovs, "HGCF: Hyperbolic graph convolution networks for collaborative filtering," in *Proc. Web Conf.*, Apr. 2021, pp. 593–601.
[51] L. Zhang and N. Wu, "HGCC: Enhancing hyperbolic graph convolution networks on heterogeneous collaborative graph for recommendation," 2023, *arXiv:2304.02961*.
[52] M. Yang, M. Zhou, J. Liu, D. Lian, and I. King, "HRCF: Enhancing collaborative filtering via hyperbolic geometric regularization," in *Proc. ACM Web Conf.*, 2022, pp. 2462–2471.
[53] H. Wang, D. Lian, H. Tong, Q. Liu, Z. Huang, and E. Chen, "HyperSoRec: Exploiting hyperbolic user and item representations with multiple aspects for social-aware recommendation," *ACM Trans. Inf. Syst.*, vol. 40, no. 2, pp. 1–28, Apr. 2022.
[54] S. Bai, F. Zhang, and P. H. S. Torr, "Hypergraph convolution and hypergraph attention," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107637.
[55] O. Ganea, G. Bécigneul, and T. Hofmann, "Hyperbolic neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 5350–5360.
[56] J. Li et al., "MINER: Multi-interest matching network for news recommendation," in *Proc. Findings Assoc. Comput. Linguistics*, 2022, pp. 343–352.
[57] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," in *Proc. 31st Int. Conf. neural Inf. Process. Syst.*, 2017, pp. 972–981.
[58] W. Guo et al., "Deep graph convolutional networks with hybrid normalization for accurate and diverse recommendation," in *Proc. 3rd Workshop Deep Learn. Pract. High-Dimensional Sparse Data KDD*, 2021, pp. 1–9.
[59] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.
[60] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.
[61] X. Xia, H. Yin, J. Yu, Y. Shao, and L. Cui, "Self-supervised graph co-training for session-based recommendation," in *Proc. 30th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2021, pp. 2180–2190.
[62] A. Li, Z. Cheng, F. Liu, Z. Gao, W. Guan, and Y. Peng, "Disentangled graph neural networks for session-based recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 7870–7882, Aug. 2023.
[63] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.

Tongcun Liu received the Ph.D. degree in computer science from Beijing University of Posts and Telecommunications, Beijing, China, in 2019.

He is currently a Lecturer with the School of Mathematics and Computer Science, Zhejiang A&F University, Hangzhou, China. His research interests include data mining, recommender systems, user behavior modeling, and predictive analytics.



Xukai Bao received the master's degree from Beijing University of Posts and Telecommunications, Beijing, China, in 2023.

His research interests focus on recommender systems and financial time series.

Jiaxin Zhang is currently pursuing the master's degree in computer science and technology with Beijing University of Posts and Telecommunications, Beijing, China.

His research focuses on recommendation systems, self-supervised learning, and tackling challenges associated with long-tail distributions.

Kai Fang (Member, IEEE) was born in Zhejiang, China, in 1992. He received the master's degree from Zhejiang University of Technology, Hangzhou, China, in 2017, and the Ph.D. degree in computer technology and application from Macau University of Science and Technology, Macau, China, in 2023.

He is currently a Professor with Zhejiang A&F University, Hangzhou. His research interests include the IoT, IoV, computational social science, and deep learning.

Hailin Feng (Member, IEEE) received the Ph.D. degree in computer science from the University of Science and Technology of China, Hefei, China, in June 2007.

Since 2007, he has been with the School of Mathematics and Computer Science, Zhejiang A& F University, Hangzhou, China. From 2013 to 2014, he was a Visiting Professor at Forest Products Laboratory, United States Department of Agriculture (USDA), WA, USA. He is currently a Professor with the School of Mathematics and Computer Science. His main interests include computer vision, intelligent information processing, and the Internet of Things.